

Construction of Energy-Stable Projection-Based Reduced Order Models

Irina Kalashnikova^a, Matthew F. Barone^b, Srinivasan Arunajatesan^b, Bart G. van Bloemen Waanders^c

^a*Computational Mathematics Department, Sandia National Laboratories, P.O. Box 5800, MS 1320, Albuquerque, NM 87185-1320.*

^b*Aerosciences Department, Sandia National Laboratories, P.O. Box 5800, MS 0825, Albuquerque, NM 87185-0825.*

^c*Optimization and Uncertainty Quantification Department, Sandia National Laboratories, P.O. Box 5800, MS 1318, Albuquerque, NM 87185-1318.*

Abstract

An approach for building energy-stable Galerkin reduced order models (ROMs) for linear hyperbolic or incompletely parabolic systems of partial differential equations (PDEs) using continuous projection is developed. This method is an extension of earlier work by the authors specific to the equations of linearized compressible inviscid flow. The key idea is to apply to the PDEs a transformation induced by the Lyapunov function for the system, and to build the ROM in the transformed variables. For linear problems, the desired transformation is induced by a special inner product, termed the “symmetry inner product”, which is derived herein for several systems of physical interest. Connections are established between the proposed approach and other stability-preserving model reduction methods, giving the paper a review flavor. More specifically, it is shown that a discrete counterpart of this inner product is a weighted L^2 inner product obtained by solving a Lyapunov equation, first proposed by Rowley *et al.* and termed herein the “Lyapunov inner product”. Comparisons between the symmetry inner product and the Lyapunov inner product are made, and the performance of ROMs constructed using these inner products is evaluated on several benchmark test cases.

Keywords: Reduced order model (ROM), proper orthogonal decomposition (POD)/Galerkin projection, linear hyperbolic/incompletely parabolic systems, linear time-invariant (LTI) systems, numerical stability, Lyapunov equation.

1. Introduction

Numerous modern-day engineering problems require the simulation of complex systems with tens of millions or more unknowns. Despite improved algorithms and the availability of massively parallel computing resources, “high-fidelity” models are, in practice, often too computationally expensive for use in a design or analysis setting. The continuing push to incorporate into modeling efforts the quantification of uncertainties, critical to many science and engineering applications, can present an intractable computational burden due to the high-dimensional systems that arise. This

situation has prompted researchers to develop reduced order models (ROMs): models constructed from high-fidelity simulations that retain the essential physics and dynamics of their corresponding full order models (FOMs), but have a much lower computational cost. Since ROMs are, by construction, small, they can enable uncertainty quantification (UQ) as well as on-the-spot decision making and/or control.

In order to serve as a useful predictive tool, a ROM should possess the following properties: consistency (with respect to its corresponding high-fidelity model), stability, and convergence (to the

solution of its corresponding high-fidelity model). The second of these properties, namely numerical stability, is particularly important, as it is a prerequisite for studying the convergence and accuracy of a ROM. It is well-known that popular model reduction approaches known as the proper orthogonal decomposition (POD) method [25; 26; 19] and the balanced proper orthogonal decomposition (BPOD) method [31; 23] lack, in general, an *a priori* stability guarantee. In [30], Amsallem *et al.* suggest that POD ROMs constructed for linear time-invariant (LTI) systems in descriptor form tend to possess better numerical stability properties than POD ROMs constructed for LTI systems in non-descriptor form. Although heuristics such as these exist, it is in general unknown *a priori* if a ROM constructed using POD or BPOD will preserve the stability properties of the high-fidelity system from which the model was constructed. There *does* exist a model reduction technique that has a rigorous stability guarantee, namely balanced truncation [29; 10]; however, the computational cost of this method, which requires the computation and simultaneous diagonalization of infinite controllability and observability Gramians, makes balanced truncation computationally intractable for systems of very large dimensions (i.e., systems with more than 10,000 degrees of freedom [24]).

The importance of obtaining stable ROMs has been recognized in recent years by a number of authors. It is shown by Patera, Veroy and Rozza in [27; 28] that a stable ROM can be constructed using the reduced basis method. In [24], Rowley *et al.* show that Galerkin projection preserves the stability of an equilibrium point at the origin if the ROM is constructed in an “energy-based” inner product. In [6; 7], Barone *et al.* demonstrate that a symmetry transformation leads to a stable formulation for a Galerkin ROM for the linearized compressible Euler equations [6; 7] with solid wall and far-field boundary conditions. In [1], Serre *et al.* propose applying the stabilizing projection developed by Barone *et al.* in [6; 7] to a skew-symmetric system constructed by augmenting a given linear system with its adjoint system. This approach yields a

ROM that is stable at finite time even if the solution energy of the full-order model is growing.

The methods described above derive (*a priori*) a stability-preserving model reduction framework that is specific to a particular equation set. There exist, in addition to these techniques, approaches which stabilize an unstable ROM through a post-processing (*a posteriori*) stabilization step applied to an unstable algebraic ROM system. Ideally, the stabilization is such that it will only minimally modify the ROM. In [5], Amsallem *et al.* propose a method for stabilizing projection-based linear ROMs through the solution of a small-scale convex optimization problem. In [38], a set of linear constraints for the left-projection matrix, given the right-projection matrix, are derived by Bond *et al.* to yield a projection framework that is guaranteed to generate a stable ROM. An approach for stabilizing unstable ROMs for LTI systems, termed ROM stabilization via optimization-based eigenvalue reassignment, was proposed by Kalashnikova *et al.* in the recent work [55]. In this approach, the unstable eigenvalues of an unstable ROM are modified through the numerical solution of a constrained nonlinear least-squares optimization problem formulated such that the error in the stabilized ROM output is minimal. In [39], a ROM stabilization methodology that achieves improved accuracy and stability through the use of a new set of basis functions representing the small, energy-dissipation scales of turbulent flows is derived by Balajewicz *et al.* In [35], Zhu *et al.* derive some large eddy simulation (LES) closure models for POD ROMs for the incompressible Navier-Stokes equations, and demonstrate numerically that the inclusion of these LES terms yields a ROM with increased numerical stability (albeit at the sacrifice of consistency of the ROM with respect to the direct numerical simulation (DNS) from which the ROM is constructed).

In this article, several approaches to building stable ROMs for linear systems, both in the continuous as well as in the discrete projection setting, are presented, connected and extended. The article has

a review flavor, but contains several new contributions, most notably the following:

- The energy-stable continuous projection ROM method developed specifically for the equations of linearized compressible inviscid flow in [6; 7] is extended to generic systems of PDEs of the hyperbolic or incompletely parabolic type.
- A stability preserving symmetry inner product is derived for several physical systems (the wave equation, the linearized shallow water equations, the linearized compressible Euler equations, the linearized compressible Navier-Stokes equations).
- Connections between the proposed energy-stable continuous projection method and other model reduction techniques with an *a priori* stability guarantee, e.g., a discrete projection approach involving a Lyapunov equation-based inner product introduced by Rowley *et al.* in [24], are established using the concept of energy stability.
- Numerical studies evaluating the performance of ROMs constructed in the energy inner products described herein are performed.

The remainder of this paper is organized as follows. The first part consists of some preliminaries: projection-based model reduction (in particular, the POD¹/Galerkin method) is overviewed (Section 2), and several notions of stability (energy-stability, Lyapunov stability, asymptotic stability, exponential stability, time-stability) are defined (Section 3). Attention is then turned to the construction of energy-stable ROMs for linear systems of PDEs using continuous projection (Section 4). The energy-stability preserving model reduction

¹For concreteness, it is assumed herein that the reduced basis is constructed via the POD method, as the POD is a popular method for computing reduced bases that is feasible even for very large systems but can give rise to unstable ROMs. It is emphasized that the energy-stability results discussed herein hold for *any* choice of reduced basis, not just the POD basis, however.

approach developed specifically for the equations of linearized compressible inviscid flow in [6; 7] is generalized. Examples of this inner product are given for several systems of physical interest, and some numerical results are presented. Next, it is shown that a certain transformation applied to a generic linear hyperbolic or incompletely parabolic set of PDEs and induced by the Lyapunov function for these equations will yield a Galerkin ROM that is stable for *any* choice of reduced basis. It is then shown that, for many PDEs, the desired transformation is induced by a special weighted L^2 inner product, termed the “symmetry inner product”. It is also demonstrated that a discrete weighted L^2 inner product first derived by Rowley *et al.* in [24] and termed herein the “Lyapunov inner product” is a discrete counterpart of the symmetry inner product. The weighting matrix that defines the Lyapunov inner product can be computed in a black-box fashion for a stable LTI system arising from the discretization of a linear system of PDEs in space. Numerical studies of POD ROMs constructed in the Lyapunov inner product are performed. A unifying summary of the energy-stability preserving model reduction approaches described within this paper is given Section 6, along with some conclusions. It is anticipated that this discussion will aid the reader in selecting the most appropriate model reduction methodology for his/her application.

2. Projection-based model reduction

In this section, several approaches to building projection-based reduced order models are reviewed. Attention is restricted to LTI systems. A system is called time-invariant if the output response for a given input does not depend on when that input is applied [15].

At the continuous level, an LTI system can be represented by a PDE (or system of PDEs) of the form

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathcal{L}(\mathbf{x}(t)) + \mathcal{L}_c(\mathbf{u}(t)), \\ \mathbf{y}(t) &= \mathcal{L}_o(\mathbf{x}(t)),\end{aligned}\tag{1}$$

in an open bounded domain Ω , subject to some

boundary conditions on the boundary of Ω , denoted by $\partial\Omega$, written abstractly as

$$\mathcal{L}_b(\mathbf{x}(t)) = \mathbf{g}(t). \quad (2)$$

Here, t denotes time, $\mathbf{x} \in \mathbb{R}^n$ is called the state vector, $\mathbf{u} \in \mathbb{R}^p$ represents the vector of control variables, $\mathbf{y} \in \mathbb{R}^q$ is the measured signal or output, and the ‘ \cdot ’ symbol denotes differentiation with respect to time, i.e., $\dot{\mathbf{x}} \equiv \frac{\partial \mathbf{x}}{\partial t}$. The operator $\mathcal{L} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a smooth linear spatial-differential operator, i.e.,

$$\mathcal{L} \equiv \sum_{|\mathbf{v}| \leq d} \mathbf{A}_{\mathbf{v}}(t) \partial_{v_1 \dots v_d}^{(|\mathbf{v}|)} \quad (3)$$

where $\mathbf{A}_{\mathbf{v}} \in \mathbb{R}^{n \times d}$, $\mathbf{v} = (v_1, \dots, v_d)$ is a multi-index, $|\mathbf{v}| = \sum_{i=1}^d v_i$, and $\partial_{v_1 \dots v_d}^{(|\mathbf{v}|)}$ denotes the i^{th} derivative with respect to variables v_1, \dots, v_d , for $i = 1, \dots, d$, where $d \in \mathbb{N}$. The boundary operator $\mathcal{L}_b : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a similar smooth linear spatial-differential operator, and $\mathbf{g} \in \mathbb{R}^n$ is a smooth function specifying the boundary data. The boundary conditions (2) can be of the Dirichlet, Neumann or Robin type, or a combination of these three types. It is assumed that the boundary conditions (2) are selected such that the resulting initial boundary value problem ((1) with boundary conditions (2) and an initial condition) is well-posed. The operators $\mathcal{L}_c : \mathbb{R}^n \rightarrow \mathbb{R}^n$ and $\mathcal{L}_o : \mathbb{R}^n \rightarrow \mathbb{R}^q$ are smooth linear mappings. The abstract operators \mathcal{L} , \mathcal{L}_b , \mathcal{L}_c and \mathcal{L}_o are introduced to keep the discussion as general as possible, and used in subsequent analysis.

Suppose the PDE system (1) has been discretized in space using some numerical scheme, e.g., the finite element method. The result will be a semi-discrete LTI system of the form:

$$\begin{aligned} \dot{\mathbf{x}}_N(t) &= \mathbf{A}\mathbf{x}_N(t) + \mathbf{B}\mathbf{u}_P(t) \\ \mathbf{y}_{QN}(t) &= \mathbf{C}\mathbf{x}_N(t). \end{aligned} \quad (4)$$

Here, $\mathbf{x}_N \in \mathbb{R}^N$ is the discretized state vector, $\mathbf{u}_P \in \mathbb{R}^P$ is the discretized vector of control variables, and $\mathbf{y}_{QN} \in \mathbb{R}^Q$ is the discretized output; $\mathbf{A} \in \mathbb{R}^{N \times N}$, $\mathbf{B} \in \mathbb{R}^{N \times P}$ and $\mathbf{C} \in \mathbb{R}^{Q \times N}$ are constant matrices (in particular, they are not functions of time t).

The general approach to projection-based model reduction consists of three steps, described below.

Step 1: Calculation of reduced trial and test bases, denoted by $\Phi_M = (\phi_1, \dots, \phi_M)$ and $\Psi_M = (\psi_1, \dots, \psi_M)$ respectively, each of order M , with $M \ll N$.

Step 2: Approximation of the solution to (1) by

$$\mathbf{x}(t) \approx \sum_{i=1}^M x_{M,i}(t) \phi_i = \Phi_M \mathbf{x}_M(t), \quad (5)$$

where $x_{M,i}(t)$ are the unknown ROM coefficients or modal amplitudes, to be determined in solving the ROM.

Step 3: Substitution of the approximation (5) into the governing system ((1) or (4)) and projection of this system onto the reduced test basis.

The result of this procedure is a “small” (size $M \ll N$) dynamical system that, for a suitable choice of reduced bases, accurately describes the dynamics of the full order system for some set of conditions. The reduced bases $\Phi_M \in \mathbb{R}^{N \times M}$ and $\Psi_M \in \mathbb{R}^{N \times M}$ are functions of space but not time, and are assumed to have full column rank. In the case that $\Psi_M \neq \Phi_M$, the projection is referred to as a Petrov-Galerkin projection. Otherwise, if $\Psi_M = \Phi_M$, the projection is referred to as a Galerkin projection. This terminology is introduced here as it will be shown later that the energy-stable model reduction approaches derived in this work are effectively Petrov-Galerkin methods.

2.1. Calculation of the reduced bases (Step 1)

There exist a number of approaches for calculating the reduced basis modes (*Step 1* of the model reduction), e.g., the POD method [25; 26; 19], the BPOD method [31; 23], the balanced truncation method [29; 10], the reduced basis method [27; 28]; also methods based on goal-oriented bases [21], generalized eigenmodes [37], and Koopman modes [40]. Attention is restricted here to the POD basis, but it is noted that the energy-stability results derived in this paper hold for *any* choice of reduced basis. The reason for the choice of the

POD reduced basis is two-fold. First, the POD is a widely used approach for computing efficient bases for dynamical systems. Moreover, ROMs constructed via the POD/Galerkin method lack in general an *a priori* stability guarantee (meaning POD/Galerkin ROMs would benefit from stability-preserving model reduction approaches such as those developed herein).

Discussed in detail in Lumley [16] and Holmes *et al.* [19], POD is a mathematical procedure that, given an ensemble of data and an inner product, denoted generically by (\cdot, \cdot) , constructs a basis for the ensemble. This basis is optimal in the sense that it describes more energy (on average) of the ensemble in the chosen inner product than any other linear basis of the same dimension M . The ensemble $\{\mathbf{x}^k : k = 1, \dots, K\}$ is typically a set of K instantaneous snapshots of a numerical solution field, taken for K values of a parameter of interest, or at K different times. Mathematically, POD seeks an M -dimensional ($M \ll K$) subspace spanned by the set $\{\phi_i\}$ such that the projection of the difference between the ensemble \mathbf{x}^k and its projection onto the reduced subspace is minimized on average. It is a well-known result [6; 19; 34; 33] that the solution to the POD optimization problem reduces to the eigenvalue problem

$$\mathbf{R}\phi = \lambda\phi, \quad (6)$$

where \mathbf{R} is a positive semi-definite matrix with its (i, j) entry given by $R_{ij} = \frac{1}{K} (\mathbf{x}^i, \mathbf{x}^j)$ for $1 \leq i, j \leq K$. It can be shown [19; 16] that the set of M eigenfunctions, or POD modes, $\{\phi_i : i = 1, \dots, M\}$ corresponding to the M largest eigenvalues of \mathbf{R} is precisely the desired basis. This is the so-called “method of snapshots” for computing a POD basis [25].

2.2. Approximation of solution in reduced basis (Step 2)

Once the reduced basis is computed, the solution $\mathbf{x}(t)$ is approximated as a linear combination of the reduced basis modes (5) (Step 2). Given this approximation, the following error formula can be

shown for the POD [19; 34]:

$$\frac{1}{K} \sum_{i=1}^K \left\| \mathbf{x}^i - \sum_{j=1}^M (\mathbf{x}^i, \phi_j) \phi_j \right\|^2 = \sum_{k=M+1}^J \lambda_k, \quad (7)$$

where $J = \dim(\{\mathbf{x}^1, \dots, \mathbf{x}^K\})$, and where $\lambda_1 \geq \dots \geq \lambda_J > 0$ are the positive eigenvalues of the operator \mathbf{R} (6).

Typically, the size of the reduced basis is chosen based on an energy criterion. That is, M is selected to be the minimum integer such that

$$E_{POD}(M) \geq \text{tol}, \quad (8)$$

where $0 \leq \text{tol} \leq 1$ represents the snapshot energy represented by the POD basis, and

$$E_{POD}(M) \equiv \frac{\sum_{i=1}^M \lambda_i}{\sum_{i=1}^K \lambda_i}. \quad (9)$$

2.3. Projection (Step 3)

There are two approaches for performing Step 3 of the model reduction: continuous and discrete projection. These approaches are described, as well as compared and contrasted, in the present subsection. Stability-preserving methods for constructing ROMs using these approaches will be detailed in Sections 4 and 5.

2.3.1. Model reduction via continuous projection

In the continuous projection approach [6; 7], the continuous system of PDEs (1) is projected onto a continuous representation (discussed in more detail below) of the reduced test basis $\{\psi_i\}_{i=1}^M \in \mathbb{R}^n$ in a continuous inner product (\cdot, \cdot) , for example, the usual L^2 inner product²

$$(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) = \int_{\Omega} \mathbf{x}^{(1)T} \mathbf{x}^{(2)} d\Omega, \quad (10)$$

where the $x_{M,i}(t)$ are the unknown ROM coefficients or modal amplitudes (so that $\mathbf{x}_M^T \equiv$

²Weighted variants of the L^2 inner product are considered later in this work.

$(x_{M,1}, \dots, x_{M,M})$, to be determined in solving the ROM dynamical system (derived below).

Substituting (5) into (1), the following is obtained

$$\begin{aligned}\sum_{i=1}^M \dot{x}_{M,i}(t) \phi_i &= \mathcal{L}(\sum_{i=1}^M x_{M,i}(t) \phi_i) \\ &+ \mathcal{L}_c(\mathbf{u}(t)), \\ \mathbf{y}_{QM}(t) &= \mathcal{L}_o(\sum_{i=1}^M x_{M,i}(t) \phi_i),\end{aligned}\quad (11)$$

where $\mathbf{y}_{QM}(t)$ is the reduced approximation of the output.

Next, a reduced test basis $\{\psi_i\}_{i=1}^M \in \mathbb{R}^n$ is introduced, and the system of PDEs (11) is projected onto a continuous representation of the reduced test basis modes ψ_j for $j = 1, 2, \dots, M$ in the inner product (\cdot, \cdot) to yield

$$\begin{aligned}\sum_{i=1}^M \dot{x}_{M,i}(t) (\psi_j, \phi_i) &= (\psi_j, \mathcal{L}(\sum_{i=1}^M x_{M,i}(t) \phi_i)) \\ &+ (\psi_j, \mathcal{L}_c(\mathbf{u}(t))), \\ \mathbf{y}_{QM}(t) &= \mathcal{L}_o(\sum_{i=1}^M x_{M,i}(t) \phi_i),\end{aligned}\quad (12)$$

for $j = 1, 2, \dots, M$. Typically, the reduced trial and test bases ϕ_i and ψ_j are chosen to be orthonormal in the inner product (\cdot, \cdot) , so that $(\psi_j, \phi_i) = \delta_{ij}$, where δ_{ij} denotes the Kronecker delta function. Invoking this property, as well as the linearity property of the operators \mathcal{L} and \mathcal{L}_o , (12) simplifies to

$$\begin{aligned}\dot{x}_{M,j}(t) &= \sum_{i=1}^M x_{M,i}(t) (\psi_j, \mathcal{L}(\phi_i)) \\ &+ (\psi_j, \mathcal{L}_c(\mathbf{u}(t))), \\ \mathbf{y}_{QM}(t) &= \sum_{i=1}^M x_{M,i}(t) \mathcal{L}_o(\phi_i),\end{aligned}\quad (13)$$

for $j = 1, 2, \dots, M$. The equations (13) define a set of M time-dependent ODEs for the modal amplitudes $x_{M,i}(t)$ in (5).

Note that, since Step 1 of the model reduction (Section 2.1) yields a *discrete*-valued basis Φ_M , before applying the continuous projection approach, this basis needs to be represented using a set of continuous basis functions. One way to do this is to cast the discrete-valued POD modes as a collection of continuous finite elements. This procedure is outlined here, and described in more detail in Section 4.4.1. Assume without loss of generality that the solution of the POD eigenvalue problem (6) gives a set of basis vectors defined at the nodes of an associated mesh that can be broken up into n_{el} disjoint

finite elements Ω_e such that $\cup_{e=1}^{n_{el}} \Omega_e = \Omega$. Assume also (without loss of generality) that the continuous solution to (11) is scalar-valued. Let $\phi_i|_{\Omega_e^j}$ denote the value of the i^{th} basis function at the j^{th} node of element Ω_e for $i = 1, \dots, M$ and $j = 1, \dots, nn$, where nn is the number of nodes of element Ω_e . Then, the finite element representation of the vector ϕ_i in the element Ω_e is:

$$\phi_i^e = \sum_{j=1}^{nn} N_j(\mathbf{x}) \phi_i|_{\Omega_e^j} \quad (14)$$

where $N_j(\mathbf{x})$ denotes the j^{th} finite element shape function. Assembling the set of functions ϕ_i^e over the full set of elements $\{\Omega_e\}_{e=1}^{n_{el}}$ gives a continuous representation of ϕ_i . A similar procedure can be applied to obtain a continuous representation of the test basis Ψ_M . The integrals in (13) can then be evaluated numerically through the use of numerical quadrature. If the integrands in (13) are polynomials, it is possible to select a quadrature rule to evaluate the integrals exactly.

2.3.2. Model reduction via discrete projection

In the discrete projection approach, the FOM ODE system (4) (the PDE system discretized in space) is projected onto a discrete reduced test basis in a discrete inner product. Suppose this discrete inner product is the following weighted L^2 inner product:

$$(\mathbf{x}_N^{(1)}, \mathbf{x}_N^{(2)})_{\mathbf{P}} = \mathbf{x}_N^{(1)T} \mathbf{P} \mathbf{x}_N^{(2)}, \quad (15)$$

where $\mathbf{P} \in \mathbb{R}^{N \times N}$ is a symmetric positive-definite matrix. Let $\Phi_M \in \mathbb{R}^{N \times M}$ and $\Psi_M \in \mathbb{R}^{N \times M}$ denote the reduced trial and reduced test bases for (4), respectively. Assume these matrices have full column rank, and are orthonormal in the inner product (15), so that $\Psi_M^T \mathbf{P} \Phi_M = \mathbf{I}_M$, where \mathbf{I}_M denotes the $M \times M$ identity matrix. The first step in constructing a ROM for (4) using discrete projection is to approximate the solution $\mathbf{x}_N(t)$ by (5). Substituting (5) into (4), and projecting this system onto the reduced test basis, the following $M \times M$ LTI system is obtained:

$$\begin{aligned}\dot{\mathbf{x}}_M(t) &= \mathbf{A}_M \mathbf{x}_M(t) + \mathbf{B}_M \mathbf{u}_P(t), \\ \mathbf{y}_{QM}(t) &= \mathbf{C}_M \mathbf{x}_M(t),\end{aligned}\quad (16)$$

where

$$\begin{aligned}\mathbf{A}_M &= \mathbf{\Psi}_M^T \mathbf{P} \mathbf{A} \mathbf{\Phi}_M, \\ \mathbf{B}_M &= \mathbf{\Psi}_M^T \mathbf{P} \mathbf{B}, \\ \mathbf{C}_M &= \mathbf{C} \mathbf{\Phi}_M,\end{aligned}\tag{17}$$

and where \mathbf{y}_{QM} is a reduced approximation of the output.

2.3.3. Continuous vs. discrete projection

In the majority of applications of reduced order modeling, the discrete projection approach is employed in constructing the ROM. This discrete approach has the advantage that boundary condition terms present in the discretized equation set are often (depending on the implementation) inherited by the ROM; that is, the ROM solution will satisfy the boundary conditions of the FOM. Certain properties of the numerical scheme used to solve the full equations may be inherited by the ROM as well. The discrete approach can be black-box, at least for linear systems of the form (4): it operates on the matrices \mathbf{A} , \mathbf{B} and \mathbf{C} , so that access to the high-fidelity code that was used to generate these matrices or the governing PDEs is not required provided these matrices can be written out from the high-fidelity code. In contrast, the continuous projection approach is tied to the governing PDEs – the continuous problem (1) needs to be translated to the discrete setting, e.g., by interpolating the reduced basis modes and evaluating the continuous inner products in (13) using a numerical quadrature [6]. Although the continuous approach is inherently an embedded method, its similarity to spectral numerical approximation methods allows the use of analysis techniques employed by the spectral methods community [36; 7].

Which of the two projection approaches described above (continuous vs. discrete projection) is preferred depends on the application and the type of model reduction approach sought (e.g., embedded vs. black-box). The discussion in the remainder of this paper is intended to aid the reader in selecting one of these approaches for his or her problem of interest.

Note that, regardless of which projection approach is used to build the ROM, the ROM dynamical system will have the form (16), as (13) has this form when written as a matrix problem. The solution to the ROM is obtained by advancing (16) forward in time using a time-integration scheme. Since the system considered here is linear, the projection terms in (13) are not time-dependent. Hence, these terms can be pre-computed and stored in the offline stage of the model reduction – in particular, they need not be re-computed at each time step of the online time-integration stage of the ROM.

3. Stability definitions

As stated in Section 1, one of the objectives of this paper is to present and establish connections between some model reduction techniques that have an *a priori* stability guarantee. Before beginning this discussion, some general definitions of stability that will be used in subsequent analysis are reviewed.

3.1. Energy-stability

The concept of energy-stability originated in the literature involving the numerical analysis of spectral and finite difference discretizations to time-dependent PDEs [47; 8; 12]. It has also appeared in the Galerkin finite element method literature, e.g., [4; 2], where the energy-method was employed to derive stable Galerkin methods for hyperbolic conservation laws. It is well-known that physical systems admit a certain energy structure. The basic idea behind building energy-stable ROMs is that a ROM constructed for such systems should preserve this energy structure. Among the authors who have explored the concept of energy-stability in the context of model reduction are Rowley *et al.* [23] and Kwasniok [3]. In [23], Rowley *et al.* introduced a family of “energy-based” inner products for the purpose of constructing stable Galerkin ROMs for fluid problems. In [3], Kwasniok recognized the role of energy conservation in ROMs of nonlinear,

incompressible fluid flow for atmospheric modeling applications, and proposed a Galerkin projection approach in which the ROM conserves turbulent kinetic energy or turbulent enstrophy.

The concept of energy-stability will be introduced in the context of a specific canonical model problem, then generalized. Consider, without loss of generality, the following scalar initial value problem, known as a Cauchy problem [20]:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathcal{L}(\mathbf{x}(t)), \quad t \geq 0 \\ \mathbf{x}(0) &= \mathbf{f}.\end{aligned}\quad (18)$$

Here, \mathcal{L} denotes a linear differential operator with constant coefficients (e.g., (3), the linear operator in (1)), $\mathbf{f} \in \mathbb{R}^n$ is the initial condition, and $\mathbf{x}(t) \in \mathbb{R}^n$ is the system state at time $t \geq 0$. No boundary conditions are given in (18), as the canonical Cauchy problem is posed on the whole real line [8]. This is equivalent to the problem being posed on a fixed domain with periodic boundary conditions. The operator \mathcal{L} is said to be semi-bounded with respect to an inner product (\cdot, \cdot) if it satisfies the following inequality for all sufficiently smooth functions $\mathbf{w} \in L^2$:

$$(\mathbf{w}, \mathcal{L}\mathbf{w}) \leq \alpha(\mathbf{w}, \mathbf{w}), \quad (19)$$

where $\alpha \in \mathbb{R}$. The following theorem (quoted from [20]) states the conditions under which the Cauchy problem (18) is well-posed.

Theorem 3.1.1 ([20], p. 70): The Cauchy problem (18) is well-posed if and only if the operator \mathcal{L} is semi-bounded with respect to an inner product (\cdot, \cdot) which corresponds to a norm equivalent to the L^2 norm.

Consider now a Galerkin approximation to (18), denoted here by \mathbf{x}_N , and satisfying

$$(\dot{\mathbf{x}}_N, \boldsymbol{\phi}) = (\mathcal{L}(\mathbf{x}_N), \boldsymbol{\phi}), \quad (20)$$

for all $\boldsymbol{\phi}$ sufficiently smooth, and suppose \mathcal{L} is semi-bounded with respect to (\cdot, \cdot) . Setting $\boldsymbol{\phi} = \mathbf{x}_N$ in (20) leads to the following energy estimate for the Galerkin approximation:

$$\frac{dE_N}{dt} \leq 2\alpha E_N, \quad (21)$$

where $E_N \equiv \frac{1}{2}||\mathbf{x}_N||^2$ denotes the energy of the Galerkin approximation \mathbf{x}_N , and $||\cdot||$ is the norm induced by the inner product (\cdot, \cdot) . Applying Gronwall's lemma ((84) in Appendix A.1) to (21) gives the inequality

$$||\mathbf{x}_N(t)|| \leq e^{\frac{1}{2}\alpha t} ||\mathbf{x}_N(0)||. \quad (22)$$

The result (22) says that the energy of the numerical solution to (20) is bounded in a way that is consistent with the behavior of the energy of the exact solution to the original differential equation (18), i.e., the numerical solution is energy-stable. This definition can be extended to a ROM LTI system of the form (16).

Definition 3.1.2 (Energy-Stability [12]): A ROM LTI system (16) is called energy-stable if

$$E_M(t) \leq e^{\alpha t} E_M(0), \quad (23)$$

for some constant $\alpha \in \mathbb{R}$, where

$$E_M \equiv \frac{1}{2}||\mathbf{x}_M||^2 \quad (24)$$

is the system energy of the ROM numerical solution \mathbf{x}_M to (16), and $||\cdot||$ is a norm equivalent to the L^2 norm.

In general, a ROM LTI system (16) is not guaranteed to satisfy Definition 3.1.2 even if the PDE system (1) is well-posed and the full order LTI system arising from the discretization of these PDEs in space (4) is stable. However, it is often possible to ensure (23) holds for the ROM LTI system through a careful selection of the reduced trial and test bases $\boldsymbol{\Phi}_M$ and $\boldsymbol{\Psi}_M$ and/or the inner product in which the projection step of the model reduction is performed (Sections 4 and 5).

3.2. Lyapunov, asymptotic and exponential stability

The concept of energy-stability can be related to classical notions of stability, namely Lyapunov stability, asymptotic stability and exponential stability. Consider an autonomous nonlinear dynamical system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^n, \quad (25)$$

where $\mathbf{f} \in \mathbb{R}^n$ is a given function, subject to some initial condition $\mathbf{x}(0) = \mathbf{x}_0$. Let \mathbf{x}_e be an equilibrium point of the system (25), meaning $\mathbf{f}(\mathbf{x}_e) = \mathbf{0}$ for all $t \geq 0$.

Definition 3.2.1 (Lyapunov, asymptotic and exponential stability) [15]: The equilibrium point \mathbf{x}_e of (25) is said to be:

- (a) *Lyapunov stable* if $\forall \varepsilon > 0$ there exists a $\delta(\varepsilon) > 0$ such that if $\|\mathbf{x}(0) - \mathbf{x}_e\| < \delta$, then $\|\mathbf{x}(t) - \mathbf{x}_e\| < \varepsilon \forall t \geq 0$.
- (b) *Asymptotically stable* if there exists a $\delta > 0$ such that if $\|\mathbf{x}(0) - \mathbf{x}_e\| < \delta$, then $\lim_{t \rightarrow \infty} \|\mathbf{x}(t) - \mathbf{x}_e\| = 0$.
- (c) *Exponentially stable* if there exist $\alpha, \beta, \delta > 0$ such that if $\|\mathbf{x}(0) - \mathbf{x}_e\| < \delta$, then $\|\mathbf{x}(t) - \mathbf{x}_e\| \leq \alpha \|\mathbf{x}(0) - \mathbf{x}_e\| e^{-\beta t} \forall t \geq 0$.

In other words, if an equilibrium point of (25) is Lyapunov stable, solutions within a distance $\delta > 0$ from it will remain a distance $\varepsilon > 0$ from it for all time; if it is asymptotically stable, solutions within this distance will eventually converge to the equilibrium; if it is exponentially stable, the solutions will not only converge, but at an exponential rate. In general, exponential stability implies asymptotic stability, and asymptotic stability implies Lyapunov stability.

The following theorem, known as the Lyapunov stability theorem [15], can be used to characterize the stability of the stability of an equilibrium point \mathbf{x}_e for (25).

Theorem 3.2.2 (Lyapunov Stability Theorem) [15]: Let V be a non-negative function on \mathbb{R}^n and let \dot{V} represent the time derivative of V along trajectories of the system dynamics (25), i.e., $\dot{V} = \frac{\partial V}{\partial \mathbf{x}} \dot{\mathbf{x}} = \frac{\partial V}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x})$. Let $B_r = B_r(\mathbf{x}_e)$ be a ball of radius r around an equilibrium point \mathbf{x}_e of (25). If there exists an $r > 0$ such that V is positive definite and \dot{V} is negative semi-definite for all $\mathbf{x} \in B_r$, then \mathbf{x}_e is Lyapunov stable.

The function V defined in Theorem 3.2.2 above is known as the Lyapunov function for the system

(25). Observe that the numerical energy E_N defined in (21) satisfies the definition of a Lyapunov function (Theorem 3.2.2) if (26) holds. Thus, if an LTI ROM (4) is energy-stable with $\alpha = 0$ (Definition 3.1.2), then the ROM is Lyapunov stable. In Section 5, it is shown how Theorem 3.1.2 can be used to define a stability-preserving inner product for building stable ROMs for (4).

The stability concepts introduced above simplify for the specific case of LTI systems of the form (4). It is straightforward to verify that for linear systems, asymptotic and exponential stability are equivalent. Moreover, the following result holds.

Theorem 3.2.3 (Asymptotic Stability Theorem for LTI Systems) [15]: An LTI system (4) is asymptotically (and exponentially) stable if and only if all the eigenvalues of \mathbf{A} have strictly negative real parts.

Theorem 3.2.3 is commonly used to check numerically (*a posteriori*) the stability of an LTI system (4) or a ROM (16) constructed for an LTI system (Section 5.2).

3.3. Time-stability

Another form of stability is what is referred to herein as “time-stability”. Essentially, a system that is time-stable is one whose solution will not “blow up” (i.e., produce an unbounded output) given a finite input and/or non-zero initial condition. For a general nonlinear system, exponential stability implies time-stability, but time-stability is a stronger notion than asymptotic stability [53]. Since exponential and asymptotic stability are equivalent for LTI systems, asymptotic stability *does* imply time-stability in this special case.

The concept of time-stability can also be defined in terms of the system energy.

Definition 3.3.1 (Time-Stability [12]): A ROM LTI system (16) is called time-stable if the numerical energy of the ROM solution is non-increasing in time, i.e., if

$$\frac{dE_N}{dt} \leq 0. \quad (26)$$

It is straightforward to demonstrate that a time-stable scheme is also energy-stable. Suppose an LTI ROM (16) is time-stable, so the ROM solution satisfies the energy estimate (26). Applying Gronwall's lemma ((84) in Appendix A.1) to this inequality, $E_N(t) \leq E_N(0)$. Thus, (23) holds with $\alpha = 0$.

In general, the converse of the above statement does not hold: energy-stability does not necessarily imply time-stability. This is to be expected. The practical implication of a ROM possessing the energy-stability property is that its numerical solution is bounded in a way that is consistent with the behavior of the exact solutions of the governing equations (1). It is possible, in general, that unstable, physical solutions to the governing PDEs exist, i.e., solutions that are unbounded as $t \rightarrow \infty$. In this case, the energy-stable ROM may also possess unstable solutions that correspond to those unstable solutions of the governing continuous equations [1].

4. Stable model reduction for LTI systems via continuous projection

In this section, an approach for building energy-stable ROMs via continuous Galerkin projection is developed for linear PDE systems of the form:

$$\dot{\mathbf{q}}' + \mathbf{A}_i \frac{\partial \mathbf{q}'}{\partial x_i} - \mathbf{K}_{ij} \frac{\partial^2 \mathbf{q}'}{\partial x_i \partial x_j} + \mathbf{G} \mathbf{q}' = \mathbf{f}, \quad (27)$$

posed in an open bounded domain Ω . In (27), $\mathbf{q}' \in \mathbb{R}^n$ denotes a vector of unknowns, $\mathbf{f} \in \mathbb{R}^n$ is a source term, \mathbf{A}_i , \mathbf{K}_{ij} and \mathbf{G} are $n \times n$ matrices, where $1 \leq i, j \leq d$, with d denoting the number of spatial dimensions, and $n \in \mathbb{N}$. The matrices \mathbf{A}_i , \mathbf{K}_{ij} and \mathbf{G} could be a function of space, but they are assumed to be steady (not a function of time t). They are also assumed to be independent of the solution \mathbf{q}' , so that (27) is linear. The so-called Einstein notation (implied summation on repeated indices) has been employed in (27) and subsequent expressions. Most linearized conservation laws, as well as many PDEs of physical interest,

can be written in the form (27). For conservation laws, a system of the form (27) is obtained by writing the solution $\mathbf{q}(\mathbf{x}, t)$ to the underlying nonlinear conservation law as a steady mean plus an unsteady fluctuation,

$$\mathbf{q}(\mathbf{x}, t) = \bar{\mathbf{q}}(\mathbf{x}) + \mathbf{q}'(\mathbf{x}, t), \quad (28)$$

and linearizing the full set of PDEs around the steady mean $\bar{\mathbf{q}}$ to yield a system of the form (27) for $\mathbf{q}'(\mathbf{x}, t)$. In this case, the matrices \mathbf{A}_i , \mathbf{K}_{ij} and \mathbf{G} appearing in (27) are functions of $\bar{\mathbf{q}}$ and its gradients. If $\mathbf{K}_{ij} = \mathbf{0} \forall i, j$, (27) is known as a hyperbolic system [14]. An example of a system of this form is the linearized compressible Euler system. A method for constructing energy-stable ROMs specifically for the compressible Euler system using continuous Galerkin projection was presented in [6; 7], and is extended to generic systems of the form (27) herein. Otherwise, if $\mathbf{K}_{ij} \neq \mathbf{0}$, (27) is known as an incompletely parabolic system [14]. A canonical example of such a system is the linearized compressible Navier-Stokes system.

Before presenting an approach for building stable ROMs for (27) using continuous Galerkin projection, some discussion of the assumptions required for these stability results is in order.

The first assumption warranting some discussion is smoothness. In general, the analysis below assumes that the solution to (27) is in the Sobolev space of first order, $H^1(\Omega)$ and that the matrices \mathbf{A}_i and \mathbf{K}_{ij} are once and twice differentiable with respect to the spatial variable, respectively. Note that in order to obtain (27) from a nonlinear conservation law through a linearization around a steady base state (28), it is required that the base flow linearized around is differentiable.

To complete the problem definition, it is necessary to specify some boundary conditions for the system (27). A detailed discussion of well-posed boundary conditions for hyperbolic and incompletely parabolic systems of the form (27) is given in [8]. The key result shown therein is that it is always possible to find a set of boundary conditions such that the problem (27) is well-posed even

in the hyperbolic limit, that is, as $\mathbf{K}_{ij} \rightarrow \mathbf{0}$ (Theorem 3.3 in [8]). Moreover, the number of Dirichlet boundary conditions required for well-posedness depends on the signs of the eigenvalues of the matrix

$$\mathbf{A}_n \equiv \mathbf{A}_i n_i, \quad (29)$$

on a boundary $\partial\Omega$ having an outward-facing unit normal \mathbf{n} . Most often, boundary conditions are formulated in the so-called characteristic variables [7; 8], given by:

$$\mathbf{w}' = \mathbf{S}_n^{-1} \mathbf{q}', \quad (30)$$

where \mathbf{S}_n is the matrix that diagonalizes \mathbf{A}_n :

$$\mathbf{A}_n = \mathbf{S}_n \mathbf{\Lambda}_n \mathbf{S}_n^{-1}, \quad (31)$$

where $\mathbf{\Lambda}_n$ is a diagonal matrix containing the eigenvalues of \mathbf{A}_n ³.

The subsequent stability proofs for ROMs constructed using continuous Galerkin projection for systems of the form (27) will assume the specific scenario in which the \mathbf{A}_n has eigenvalues that are all the same sign on the boundaries of $\partial\Omega$. More specifically, it will be assumed that $\partial\Omega$ can be partitioned as $\partial\Omega = \partial\Omega_I \cup \partial\Omega_O$, where $\partial\Omega_I \cap \partial\Omega_O = \emptyset$ and \mathbf{A}_n is negative definite on $\partial\Omega_I$ and positive definite on $\partial\Omega_O$, so that well-posed boundary conditions are all-Dirichlet boundary conditions on $\partial\Omega_I$ and no boundary conditions on $\partial\Omega_O$ [8]. This scenario, and the boundary conditions it leads to, arise in many physically-relevant problems, e.g., fluid problems with a supersonic inflow and supersonic outflow [7], and enables a clean, generic stability analysis of the ROM including boundary condition terms.

It will also be assumed (again, to enable a clean, generic stability analysis of the ROM) that the all-Dirichlet boundary condition on $\partial\Omega_I$ is steady (not a function of time), i.e.,

$$\mathbf{q}(\mathbf{x}, t) = \mathbf{g}(\mathbf{x}), \text{ on } \partial\Omega_I, \quad (32)$$

³It can be shown that \mathbf{A}_n is diagonalizable for hyperbolic and incompletely parabolic systems [8].

where $\mathbf{g}(\mathbf{x})$ is a function of boundary data. It is straightforward to see that if the base flow $\bar{\mathbf{q}}(\mathbf{x})$ is selected such that it satisfies the boundary condition (32), the relevant boundary conditions on the fluctuation $\mathbf{q}'(\mathbf{x}, t)$ are of the homogeneous Dirichlet type:

$$\mathbf{q}'(\mathbf{x}, t) = \mathbf{0}, \text{ on } \partial\Omega_I. \quad (33)$$

It is equally straightforward to show that if the snapshots for $\mathbf{q}'(\mathbf{x}, t)$ satisfy the boundary condition (33), the POD modes obtained from these snapshots will satisfy this boundary condition as well.

Lastly, in the case $\mathbf{K}_{ij} \neq \mathbf{0}$ (e.g., a viscous flow problem), it will be assumed that the viscous boundary conditions imposed are well-posed, i.e., they satisfy the well-known well-posedness conditions derived in [8].

A stability analysis for boundary conditions arising from a scenario in which \mathbf{A}_n does not have strictly positive or negative eigenvalues (e.g., subsonic inflow, subsonic outflow in a fluid context) and/or the boundary conditions are time-dependent is often possible, but requires a case-by-case examination of the boundary terms that arise in the proofs of Theorem 4.1.1 and Corollary 4.2.1. This study was one undertaken in earlier work by the authors for the specific case of linearized inviscid compressible flow in some earlier work [7; 6]. The boundary conditions can be implemented in a ROM constructed via continuous Galerkin projection using either a weak formulation or the penalty method. For completeness, these approaches are detailed in Appendix A.4.

4.1. A stabilizing transformation

Suppose there exists a transformation

$$\begin{aligned} T : \mathbb{R}^n &\rightarrow \mathbb{R}^n, \\ \mathbf{q}' &\rightarrow \mathbf{v}', \end{aligned} \quad (34)$$

such that $\mathbf{v}' = \mathbf{v}'(T\mathbf{q}')$, and such that in the new variables \mathbf{v}' , the system (27) has the form

$$\dot{\mathbf{v}}' + \mathbf{A}_i^S \frac{\partial \mathbf{v}'}{\partial x_i} - \mathbf{K}_{ij}^S \frac{\partial^2 \mathbf{v}'}{\partial x_i \partial x_j} + \mathbf{G}^S \mathbf{v}' = \mathbf{f}^S, \quad (35)$$

where:

- *Property 1:* The matrices \mathbf{A}_i^S are symmetric for all $1 \leq i \leq d$.
- *Property 2:* The matrices \mathbf{K}_{ij}^S are symmetric for all $1 \leq i, j \leq d$.
- *Property 3:* The augmented viscosity matrix:

$$\mathbf{K}^S \equiv \begin{pmatrix} \mathbf{K}_{11}^S & \cdots & \mathbf{K}_{1d}^S \\ \vdots & \ddots & \vdots \\ \mathbf{K}_{d1}^S & \cdots & \mathbf{K}_{dd}^S \end{pmatrix} \quad (36)$$

is positive semi-definite.

Theorem 4.1.1: Suppose a ROM for (35) on an open bounded domain Ω with a smooth boundary $\partial\Omega = \partial\Omega_I \cup \partial\Omega_O$, $\partial\Omega_I \cap \partial\Omega_O = \emptyset$ is constructed for \mathbf{v}' using continuous Galerkin projection in the $L^2(\Omega)$ inner product. Suppose \mathbf{A}_n^S (29) is negative definite on $\partial\Omega_I$ (e.g., $\partial\Omega_I$ is a supersonic inflow boundary in a fluid mechanics context), and positive definite on $\partial\Omega_O$ (e.g., $\partial\Omega_O$ is a supersonic outflow boundary in a fluid mechanics context), so that the following boundary conditions on \mathbf{v}' are well-posed:

- Homogeneous all-Dirichlet boundary conditions ($\mathbf{v}' = \mathbf{0}$) on $\partial\Omega_I$.
- No boundary conditions on $\partial\Omega_O$.

If $\mathbf{K}_{ij}^S \neq \mathbf{0}$ suppose additional viscous boundary conditions are imposed following the criteria listed in [8] such that the resulting IBVP is well-posed. Suppose the matrices in (35) satisfy *Properties 1–3* above. Let \mathbf{v}'_M denote the ROM solution to (35). Then the ROM is energy-stable with energy estimate

$$\|\mathbf{v}'_M(\cdot, T)\|_2 \leq e^{\frac{1}{2}\beta_S T} \|\mathbf{v}'_M(\cdot, 0)\|_2, \quad (37)$$

where β_S is an upper bound on the eigenvalues of the matrix

$$\mathbf{B}^S \equiv \frac{\partial \mathbf{A}_i^S}{\partial x_i} + \frac{\partial^2 \mathbf{K}_{ij}^S}{\partial x_i \partial x_j} - \mathbf{G}^S - (\mathbf{G}^S)^T. \quad (38)$$

Moreover, this energy-stability result holds for *any* choice of reduced basis.

Proof. To prove energy-stability of a ROM constructed for (35), it is necessary to bound the energy of the ROM solution to (35) with $\mathbf{f}^S = \mathbf{0}$:

$$\begin{aligned} \frac{dE_M}{dt} &= \frac{1}{2} \frac{d}{dt} \|\mathbf{v}'_M\|_2^2 \\ &= \frac{1}{2} \frac{d}{dt} (\mathbf{v}'_M, \mathbf{v}'_M) \\ &= \left(\mathbf{v}'_M, \frac{\partial \mathbf{v}'_M}{\partial t} \right) \\ &= \left(\mathbf{v}'_M, -\mathbf{A}_i^S \frac{\partial \mathbf{v}'_M}{\partial x_i} + \mathbf{K}_{ij}^S \frac{\partial^2 \mathbf{v}'_M}{\partial x_i \partial x_j} - \mathbf{G}^S \mathbf{v}'_M \right) \\ &= -\int_{\Omega} (\mathbf{v}'_M)^T \mathbf{A}_i^S \frac{\partial \mathbf{v}'_M}{\partial x_i} d\Omega + \int_{\Omega} (\mathbf{v}'_M)^T \mathbf{K}_{ij}^S \frac{\partial^2 \mathbf{v}'_M}{\partial x_i \partial x_j} d\Omega \\ &\quad - \int_{\Omega} (\mathbf{v}'_M)^T \mathbf{G}^S \mathbf{v}'_M d\Omega. \end{aligned} \quad (39)$$

Each of the terms in (39) will be bounded separately. First,

$$\begin{aligned} -\int_{\Omega} (\mathbf{v}'_M)^T \mathbf{A}_i^S \frac{\partial \mathbf{v}'_M}{\partial x_i} d\Omega &= -\frac{1}{2} \int_{\Omega} \frac{\partial}{\partial x_i} ((\mathbf{v}'_M)^T \mathbf{A}_i^S \mathbf{v}'_M) d\Omega \\ &\quad + \frac{1}{2} \int_{\Omega} (\mathbf{v}'_M)^T \frac{\partial \mathbf{A}_i^S}{\partial x_i} \mathbf{v}'_M d\Omega \\ &= -\frac{1}{2} \int_{\partial\Omega_I} (\mathbf{v}'_M)^T \mathbf{A}_i^S n_i \mathbf{v}'_M d\Gamma \\ &\quad - \frac{1}{2} \int_{\partial\Omega_O} (\mathbf{v}'_M)^T \mathbf{A}_i^S n_i \mathbf{v}'_M d\Gamma \\ &\quad + \frac{1}{2} \int_{\Omega} (\mathbf{v}'_M)^T \frac{\partial \mathbf{A}_i^S}{\partial x_i} \mathbf{v}'_M d\Omega. \end{aligned} \quad (40)$$

In (40), the property that each of the matrices \mathbf{A}_i^S is symmetric has been employed (*Property 1*). Substituting the homogeneous all-Dirichlet boundary condition on $\partial\Omega_I$ into the first integral in (40), and employing the fact that \mathbf{A}_n^S is symmetric positive definite on $\partial\Omega_O$, the following bound is obtained:

$$\begin{aligned} -\int_{\Omega} (\mathbf{v}'_M)^T \mathbf{A}_i^S \frac{\partial \mathbf{v}'_M}{\partial x_i} d\Omega &= -\frac{1}{2} \int_{\partial\Omega_O} (\mathbf{v}'_M)^T \mathbf{A}_i^S n_i \mathbf{v}'_M d\Gamma \\ &\quad + \frac{1}{2} \int_{\Omega} (\mathbf{v}'_M)^T \frac{\partial \mathbf{A}_i^S}{\partial x_i} \mathbf{v}'_M d\Omega \\ &\leq \frac{1}{2} \int_{\Omega} (\mathbf{v}'_M)^T \frac{\partial \mathbf{A}_i^S}{\partial x_i} \mathbf{v}'_M d\Omega. \end{aligned} \quad (41)$$

Next, note that:

$$\mathbf{K}_{ij}^S \frac{\partial^2 \mathbf{v}'_M}{\partial x_i \partial x_j} = \frac{\partial}{\partial x_i} \left(\mathbf{K}_{ij}^S \frac{\partial \mathbf{v}'_M}{\partial x_j} \right) - \left(\frac{\partial \mathbf{K}_{ij}^S}{\partial x_i} \frac{\partial \mathbf{v}'_M}{\partial x_j} \right). \quad (42)$$

Then,

$$\begin{aligned} \int_{\Omega} (\mathbf{v}'_M)^T \mathbf{K}_{ij}^S \frac{\partial^2 \mathbf{v}'_M}{\partial x_i \partial x_j} d\Omega &= \int_{\Omega} (\mathbf{v}'_M)^T \frac{\partial}{\partial x_i} \left(\mathbf{K}_{ij}^S \frac{\partial \mathbf{v}'_M}{\partial x_j} \right) d\Omega \\ &\quad - \int_{\Omega} (\mathbf{v}'_M)^T \frac{\partial \mathbf{K}_{ij}^S}{\partial x_i} \frac{\partial \mathbf{v}'_M}{\partial x_j} d\Omega. \end{aligned} \quad (43)$$

Again, each of the two terms in (43) will be bounded separately.

$$\begin{aligned} \int_{\Omega} (\mathbf{v}'_M)^T \frac{\partial}{\partial x_i} \left(\mathbf{K}_{ij}^S \frac{\partial \mathbf{v}'_M}{\partial x_j} \right) d\Omega &= - \int_{\Omega} \frac{\partial \mathbf{v}'_M}{\partial x_i}^T \mathbf{K}_{ij}^S \frac{\partial \mathbf{v}'_M}{\partial x_j} d\Omega \\ &\quad + \int_{\partial\Omega_I} (\mathbf{v}'_M)^T \mathbf{K}_{ij}^S \frac{\partial \mathbf{v}'_M}{\partial x_j} n_i d\Gamma \\ &\quad + \int_{\partial\Omega_O} (\mathbf{v}'_M)^T \mathbf{K}_{ij}^S \frac{\partial \mathbf{v}'_M}{\partial x_j} n_i d\Gamma \\ &\leq \int_{\partial\Omega_I} (\mathbf{v}'_M)^T \mathbf{K}_{ij}^S \frac{\partial \mathbf{v}'_M}{\partial x_j} n_i d\Gamma \\ &\quad + \int_{\partial\Omega_O} (\mathbf{v}'_M)^T \mathbf{K}_{ij}^S \frac{\partial \mathbf{v}'_M}{\partial x_j} n_i d\Gamma, \end{aligned} \quad (44)$$

provided the matrix (36) is positive semi-definite (*Property 3*).

Now for the second term in (43):

$$\begin{aligned} &- \int_{\Omega} (\mathbf{v}'_M)^T \frac{\partial \mathbf{K}_{ij}^S}{\partial x_i} \frac{\partial \mathbf{v}'_M}{\partial x_j} d\Omega \\ &= - \frac{1}{2} \int_{\Omega} \frac{\partial}{\partial x_j} \left((\mathbf{v}'_M)^T \frac{\partial \mathbf{K}_{ij}^S}{\partial x_i} \mathbf{v}'_M \right) d\Omega \\ &\quad + \frac{1}{2} \int_{\Omega} (\mathbf{v}'_M)^T \frac{\partial^2 \mathbf{K}_{ij}^S}{\partial x_i \partial x_j} \mathbf{v}'_M d\Omega \\ &= - \frac{1}{2} \int_{\partial\Omega_I} (\mathbf{v}'_M)^T \frac{\partial \mathbf{K}_{ij}^S}{\partial x_i} n_j \mathbf{v}'_M d\Gamma \\ &\quad - \frac{1}{2} \int_{\partial\Omega_O} (\mathbf{v}'_M)^T \frac{\partial \mathbf{K}_{ij}^S}{\partial x_i} n_j \mathbf{v}'_M d\Gamma \\ &\quad + \frac{1}{2} \int_{\Omega} (\mathbf{v}'_M)^T \frac{\partial^2 \mathbf{K}_{ij}^S}{\partial x_i \partial x_j} \mathbf{v}'_M d\Omega. \end{aligned} \quad (45)$$

In (45), the property that the \mathbf{K}_{ij}^S matrices and therefore their derivatives are symmetric has been employed (*Property 2*).

Finally, (41) and (43) are substituted into (39). As shown in [8], the viscous boundary integral terms on $\partial\Omega_I$ will be negative provided the viscous boundary conditions are well-posed. The following bound is obtained:

$$\begin{aligned} \frac{1}{2} \frac{d}{dt} \|\mathbf{v}'_M\|_2^2 &\leq \frac{1}{2} \int_{\Omega} (\mathbf{v}'_M)^T \left(\frac{\partial \mathbf{A}_i^S}{\partial x_i} \right) \mathbf{v}'_M d\Omega \\ &\quad + \frac{1}{2} \int_{\Omega} (\mathbf{v}'_M)^T \frac{\partial^2 \mathbf{K}_{ij}^S}{\partial x_i \partial x_j} \mathbf{v}'_M d\Omega \\ &\quad - \frac{1}{2} \int_{\Omega} (\mathbf{v}'_M)^T \mathbf{G}^S \mathbf{v}'_M d\Omega \\ &\quad - \frac{1}{2} \int_{\Omega} (\mathbf{v}'_M)^T (\mathbf{G}^S)^T \mathbf{v}'_M d\Omega \\ &\quad - \frac{1}{2} \int_{\partial\Omega_N} (\mathbf{v}'_M)^T \mathbf{A}_i^S n_i \mathbf{v}'_M d\Gamma \\ &= \frac{1}{2} \|\mathbf{B}^S\|_2^2, \end{aligned} \quad (46)$$

where \mathbf{B}^S is given by (38). Applying Gronwall's inequality ((84) in Appendix A.1) to (46), it is found that:

$$\|\mathbf{v}'_M(\cdot, T)\|_2 \leq e^{\frac{1}{2}\beta_S T} \|\mathbf{v}'_M(\cdot, 0)\|_2, \quad (47)$$

where β_S is an upper bound on the eigenvalues of the matrix \mathbf{B}^S (38). \square

The proof of Theorem 4.1.1 is one of the new contributions of this article.

Note that, if $\mathbf{G} = \mathbf{0}$ in (27) and the \mathbf{A}_i and \mathbf{K}_{ij} matrices are spatially-constant, it follows that $\beta_S = 0$ in (47). In this case, if the ROM for (27) is constructed in the variables \mathbf{v} , the ROM will be time-stable as well as stable in the sense of Lyapunov, in addition to being energy-stable. For linearized conservation laws (e.g., the linearized shallow water equations, the linearized compressible Euler equations, the linearized compressible Navier-Stokes equations), the property that $\mathbf{G} = \mathbf{0}$ and the \mathbf{A}_i and \mathbf{K}_{ij} are spatially-constant will in general hold if the base flow is spatially uniform.

4.2. Stability-preserving “symmetry inner product” and Petrov-Galerkin connection

A key property of systems of the form (27) is that they are symmetrizable [8; 6; 7]; that is, it is possible to derive a symmetric positive-definite matrix \mathbf{H} such that:

- *Property 1**: The matrices $\mathbf{H}\mathbf{A}_i$ are symmetric for all $1 \leq i \leq d$.
- *Property 2**: The matrices $\mathbf{H}\mathbf{K}_{ij}$ are symmetric for all $1 \leq i, j \leq d$.
- *Property 3**: The augmented viscosity matrix:

$$\mathbf{K}^H \equiv \begin{pmatrix} \mathbf{H}\mathbf{K}_{11} & \dots & \mathbf{H}\mathbf{K}_{1d} \\ \vdots & \ddots & \vdots \\ \mathbf{H}\mathbf{K}_{d1} & \dots & \mathbf{H}\mathbf{K}_{dd} \end{pmatrix} \quad (48)$$

is positive semi-definite.

Since \mathbf{H} is symmetric positive-definite, the following defines a valid inner product:

$$\left(\mathbf{q}^{(1)}, \mathbf{q}^{(2)} \right)_{(\mathbf{H}, \Omega)} \equiv \int_{\Omega} \mathbf{q}^{(1)T} \mathbf{H} \mathbf{q}^{(2)} d\Omega. \quad (49)$$

Following the terminology introduced in [6; 7], the inner product (49) will be referred to as the

“symmetry inner product”. It is straightforward to see that the following corollary to Theorem 4.1.1 holds.

For a general system of PDEs, e.g., the linearized Euler equations (Example 3 in Section 4.3), the norm induced by the symmetry inner product (49) is not a recognizable “energy” quantity, but still satisfies the requisite mathematical properties for an energy analysis. For some PDEs, there is a clear connection between the energy inner product and a physical energy quantity associated with those equations, e.g., the L^2 inner product for the incompressible Navier-Stokes equations, which is identical to the kinetic energy of the solution, or the entropy inner product for the full nonlinear compressible Navier-Stokes equations, which represents the solution entropy [62].

Corollary 4.2.1: Suppose a ROM for (27) on an open bounded domain Ω with the same boundary condition assumptions as those in Theorem 4.1.1 for \mathbf{q}' on $\partial\Omega = \partial\Omega_I \cup \partial\Omega_O$, where $\partial\Omega_I \cap \partial\Omega_O = \emptyset$ is constructed using continuous Galerkin projection in the symmetry inner product (49). Suppose *Properties 1*-3** hold. Let \mathbf{q}_M denote the ROM solution to (27). Then the ROM is energy-stable with energy estimate

$$\|\mathbf{q}'_M(\cdot, T)\|_{(\mathbf{H}, \Omega)} \leq e^{\frac{1}{2}\beta_H T} \|\mathbf{q}'_M(\cdot, 0)\|_{(\mathbf{H}, \Omega)}, \quad (50)$$

where β_H is an upper bound on the eigenvalues of the matrix

$$\mathbf{B}^H \equiv \frac{\partial(\mathbf{H}\mathbf{A}_i)}{\partial x_i} + \frac{\partial^2(\mathbf{H}\mathbf{K}_{ij})}{\partial x_i \partial x_j} - \mathbf{H}\mathbf{G} - \mathbf{G}^T \mathbf{H} \quad (51)$$

Moreover, this energy-stability result holds for *any* choice of reduced basis.

Proof. Because of simple linear transformations, the proof is analogous to the proof of Theorem 4.1.1. \square

Again, in the case that $\mathbf{G} = \mathbf{0}$ and the \mathbf{A}_i , \mathbf{K}_{ij} and \mathbf{H} matrices are spatially-constant (which will occur if the base flow is uniform, i.e., not a function

of the spatial variable \mathbf{x}), it will follow from Corollary 4.2.1 that a ROM constructed in the symmetry inner product (49) will be time-stable and stable in the sense of Lyapunov, in addition to being energy-stable.

It is interesting to observe that a Galerkin projection of the governing (27) in the symmetry inner product (49) is equivalent to a Petrov-Galerkin projection. Let $\boldsymbol{\phi}_i$ for $i = 1, \dots, M$ denote the reduced trial basis vector for the solution \mathbf{q} . Performing a Galerkin projection of the equations (27) onto the modes $\boldsymbol{\phi}_k$ gives

$$\begin{aligned} \int_{\Omega} \boldsymbol{\phi}_k^T \mathbf{H} \left(\dot{\mathbf{q}}' + \mathbf{A}_i \frac{\partial \mathbf{q}'}{\partial x_i} + \mathbf{K}_{ij} \frac{\partial^2 \mathbf{q}'}{\partial x_i \partial x_j} + \mathbf{G} \mathbf{q}' \right) d\Omega \\ = \int_{\Omega} \boldsymbol{\phi}_k^T \mathbf{H} \mathbf{f} d\Omega, \end{aligned} \quad (52)$$

for $k = 1, \dots, M$. Equation (52) is equivalent to a Petrov-Galerkin projection of the equations (27) in the regular L^2 inner product

$$\begin{aligned} \int_{\Omega} \boldsymbol{\psi}_k^T \left(\dot{\mathbf{q}}' + \mathbf{A}_i \frac{\partial \mathbf{q}'}{\partial x_i} + \mathbf{K}_{ij} \frac{\partial^2 \mathbf{q}'}{\partial x_i \partial x_j} + \mathbf{G} \mathbf{q}' \right) d\Omega \\ = \int_{\Omega} \boldsymbol{\psi}_k^T \mathbf{f} d\Omega, \end{aligned} \quad (53)$$

where the reduced test basis functions are given by $\boldsymbol{\psi}_k = \mathbf{H} \boldsymbol{\phi}_k$, for all $k = 1, \dots, M$.

4.3. Examples of stability-preserving transformation and symmetry inner product for several physical systems

It is straightforward to derive the matrix \mathbf{H} that defines the symmetry inner product (49) for many problems of physical interest. This matrix has been derived herein by the authors for several hyperbolic and incompletely parabolic systems (the wave equation, the linearized shallow water equations, the linearized compressible Euler equations, and the linearized compressible Navier-Stokes equations), and is given below.

Example 1: Wave Equation

Consider the one-dimensional (1D) wave equation:

$$\ddot{u} = a^2 \frac{\partial^2 u}{\partial x^2}, \quad (54)$$

where $a \in \mathbb{R}$ denotes the wave speed, and $\ddot{u} \equiv \frac{\partial^2 u}{\partial t^2}$. (54) is a canonical PDE of the hyperbolic type. This equation can be written as a first order system

$$\dot{\mathbf{q}} = \mathbf{A} \frac{\partial \mathbf{q}}{\partial x}, \quad (55)$$

where

$$\mathbf{q} = \begin{pmatrix} \dot{u} \\ \frac{\partial u}{\partial x} \end{pmatrix}, \quad \mathbf{A} = \begin{pmatrix} 0 & a^2 \\ 1 & 0 \end{pmatrix}. \quad (56)$$

Remark that if

$$\mathbf{H} = \begin{pmatrix} 1 & 0 \\ 0 & a^2 \end{pmatrix}, \quad (57)$$

the matrix $\mathbf{H}\mathbf{A}$ is symmetric [32].

Example 2: Linearized Shallow Water Equations

Consider the linearized form of the shallow water equations:

$$\dot{\mathbf{q}}' + \mathbf{A}_i \frac{\partial \mathbf{q}'}{\partial x_i} + \mathbf{G} \mathbf{q}' = \mathbf{0}. \quad (58)$$

These equations are obtained from the full (non-linear) shallow water equations by decomposing the fluid vector $\mathbf{q}(\mathbf{x}, t)$ into a steady mean plus an unsteady fluctuation (28), and linearizing the full shallow water equations around the steady mean state $\bar{\mathbf{q}}$. If $\mathbf{q}^T = (u, v, w, \phi)$, then the convective flux matrices in the hyperbolic system (58) in three-dimensions (3D) are given by:

$$\mathbf{A}_1 = \begin{pmatrix} \bar{u} & 0 & 0 & 1 \\ 0 & \bar{u} & 0 & 0 \\ 0 & 0 & \bar{u} & 0 \\ \bar{\phi} & 0 & 0 & \bar{u} \end{pmatrix}, \quad \mathbf{A}_2 = \begin{pmatrix} \bar{v} & 0 & 0 & 0 \\ 0 & \bar{v} & 0 & 1 \\ 0 & 0 & \bar{v} & 0 \\ 0 & \bar{\phi} & 0 & \bar{v} \end{pmatrix},$$

$$\mathbf{A}_3 = \begin{pmatrix} \bar{w} & 0 & 0 & 0 \\ 0 & \bar{w} & 0 & 0 \\ 0 & 0 & \bar{w} & 1 \\ 0 & 0 & \bar{\phi} & \bar{w} \end{pmatrix}, \quad (59)$$

where ϕ denotes the local height of the fluid above the equilibrium depth, and u, v , and w are the com-

ponents of the fluid velocity vector [32]. The matrix \mathbf{G} in (58) is given by

$$\mathbf{G} = \begin{pmatrix} \frac{\partial \bar{u}}{\partial x} & \frac{\partial \bar{u}}{\partial y} & \frac{\partial \bar{u}}{\partial z} & 0 \\ \frac{\partial \bar{v}}{\partial x} & \frac{\partial \bar{v}}{\partial y} & \frac{\partial \bar{v}}{\partial z} & 0 \\ \frac{\partial \bar{w}}{\partial x} & \frac{\partial \bar{w}}{\partial y} & \frac{\partial \bar{w}}{\partial z} & 0 \\ \frac{\partial \bar{\phi}}{\partial x} & \frac{\partial \bar{\phi}}{\partial y} & \frac{\partial \bar{\phi}}{\partial z} & \nabla \cdot \bar{\mathbf{u}} \end{pmatrix}. \quad (60)$$

Each of the convective flux matrices (59) can be symmetrized by the matrix

$$\mathbf{H} = \begin{pmatrix} \bar{\phi} & 0 & 0 & 0 \\ 0 & \bar{\phi} & 0 & 0 \\ 0 & 0 & \bar{\phi} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (61)$$

Example 3: Linearized Compressible Euler Equations

Consider the linearized compressible Euler equations. These equations may be used if a compressible fluid system can be described by inviscid, small-amplitude perturbations about a steady-state mean flow. The equations are obtained from the full (non-linear) compressible Euler equations by decomposing the fluid vector $\mathbf{q}(\mathbf{x}, t)$ into a steady mean plus an unsteady fluctuation (28) and linearizing these equations around the steady mean state $\bar{\mathbf{q}}$. If $\mathbf{q}^T = (u, v, w, \zeta, p)$, where u, v and w are the three components of the velocity vector, ζ is the specific volume (the reciprocal of the density), and p is the pressure, the linearized compressible Euler equations take the form (58). In 3D, the convective flux matrices \mathbf{A}_i in the linearized compressible Euler hyperbolic system (58)

are given by:

$$\begin{aligned} \mathbf{A}_1 &= \begin{pmatrix} \bar{u} & 0 & 0 & 0 & \bar{\zeta} \\ 0 & \bar{u} & 0 & 0 & 0 \\ 0 & 0 & \bar{u} & 0 & 0 \\ -\bar{\zeta} & 0 & 0 & \bar{u} & 0 \\ \gamma\bar{p} & 0 & 0 & 0 & \bar{u} \end{pmatrix}, \\ \mathbf{A}_2 &= \begin{pmatrix} \bar{v} & 0 & 0 & 0 & 0 \\ 0 & \bar{v} & 0 & 0 & \bar{\zeta} \\ 0 & 0 & \bar{v} & 0 & 0 \\ 0 & -\bar{\zeta} & 0 & \bar{v} & 0 \\ 0 & \gamma\bar{p} & 0 & 0 & \bar{v} \end{pmatrix}, \\ \mathbf{A}_3 &= \begin{pmatrix} \bar{w} & 0 & 0 & 0 & 0 \\ 0 & \bar{w} & 0 & 0 & 0 \\ 0 & 0 & \bar{w} & 0 & \bar{\zeta} \\ 0 & 0 & -\bar{\zeta} & \bar{w} & 0 \\ 0 & 0 & \gamma\bar{p} & 0 & \bar{w} \end{pmatrix}. \end{aligned} \quad (62)$$

Here, $\gamma = C_P/C_V$ is the ratio of specific heats. The matrix \mathbf{G} in (58) has the form

$$\mathbf{G} = \begin{pmatrix} \frac{\partial \bar{u}}{\partial x} & \frac{\partial \bar{u}}{\partial y} & \frac{\partial \bar{u}}{\partial z} & \frac{\partial \bar{p}}{\partial x} & 0 \\ \frac{\partial \bar{u}}{\partial x} & \frac{\partial \bar{u}}{\partial y} & \frac{\partial \bar{u}}{\partial z} & \frac{\partial \bar{p}}{\partial y} & 0 \\ \frac{\partial \bar{u}}{\partial x} & \frac{\partial \bar{u}}{\partial y} & \frac{\partial \bar{u}}{\partial z} & \frac{\partial \bar{p}}{\partial z} & 0 \\ \frac{\partial \bar{\zeta}}{\partial x} & \frac{\partial \bar{\zeta}}{\partial y} & \frac{\partial \bar{\zeta}}{\partial z} & -\nabla \cdot \bar{\mathbf{u}} & 0 \\ \frac{\partial \bar{p}}{\partial x} & \frac{\partial \bar{p}}{\partial y} & \frac{\partial \bar{p}}{\partial z} & 0 & \gamma \nabla \cdot \bar{\mathbf{u}} \end{pmatrix}. \quad (63)$$

The reader may verify that if the linearized compressible Euler system (58) is pre-multiplied by the following symmetric positive definite matrix:

$$\mathbf{H} = \begin{pmatrix} \bar{\rho} & 0 & 0 & 0 & 0 \\ 0 & \bar{\rho} & 0 & 0 & 0 \\ 0 & 0 & \bar{\rho} & 0 & 0 \\ 0 & 0 & 0 & \alpha^2 \gamma \bar{\rho}^2 \bar{p} & \bar{\rho} \alpha^2 \\ 0 & 0 & 0 & \bar{\rho} \alpha^2 & \frac{(1+\alpha^2)}{\gamma \bar{\rho}} \end{pmatrix}, \quad (64)$$

where α is a real, non-zero parameter to yield the system, the convective flux matrices $\mathbf{H}\mathbf{A}_i$ are all symmetric [6; 7].

Example 4: Linearized Compressible Navier-Stokes Equations

Consider the 3D linearized compressible Navier-Stokes equations. These equations are appropriate

when a compressible fluid system can be described by viscous, small-amplitude perturbations about a steady-state base flow. As with the linearized shallow water equations and linearized compressible Euler equations, to derive these equations from the full (non-linear) compressible Navier-Stokes equations, the fluid vector $\mathbf{q}(\mathbf{x}, t)$ is written as the sum of a steady mean plus an unsteady fluctuation (28), and a linearization around the steady mean is performed. If the viscous work terms are neglected from the equations⁴ (appropriate, for example, in a low Mach number regime), the result is a linear incompletely parabolic system of the form (27). If the fluid vector is given by $\mathbf{q}^T = (u, v, w, T, \rho)$, where T and ρ denote the fluid temperature and density respectively, the convective and viscous flux matrices that appear in (27) are given by the expressions found in [8], and are repeated in Appendix A.5 to keep this article self-contained. The reader can verify that if the system (27) is pre-multiplied by the symmetric positive definite matrix given by

$$\mathbf{H} \equiv \begin{pmatrix} \bar{\rho} & 0 & 0 & 0 & 0 \\ 0 & \bar{\rho} & 0 & 0 & 0 \\ 0 & 0 & \bar{\rho} & 0 & 0 \\ 0 & 0 & 0 & \frac{\bar{\rho} R}{T(\gamma-1)} & 0 \\ 0 & 0 & 0 & 0 & \frac{R\bar{T}}{\bar{\rho}} \end{pmatrix}, \quad (65)$$

the “symmetrized” convective flux matrices $\mathbf{H}\mathbf{A}_i$ and diffusive flux matrices $\mathbf{H}\mathbf{K}_{ij}$ satisfy *Properties 1*–3** in Section 4.2. Here, R denotes the universal gas constant.

Note that the symmetry transformations in the examples above are not unique. For example, in [9], Abarbanel *et al.* exhibit a transformation of the

⁴To the authors’ knowledge, the viscous work terms are invariably neglected from the linearized compressible Navier-Stokes equations by researchers studying energy-stability of these equations [8; 9]. The omission of these terms is justified only in the low Mach number regime, or in the case that the base flow is uniform. The extension of the energy-stability symmetrization approach presented here to the linearized compressible Navier-Stokes equations in which the viscous work terms are retained is the subject of present research.

form (35) for the linearized compressible Navier-Stokes equations written in the primitive variables $\mathbf{q}^T = (\rho, u, v, w, p)$.

4.4. Numerical experiments

The stability-preserving model reduction approach based on continuous projection described in Sections 4.1–4.2 is now evaluated numerically on a test case involving a 2D inviscid acoustic pressure pulse in a 2D prismatic domain. The governing equations are the equations of linearized compressible flow, given in Section 4.3 (Example 3) above. Prior to showing these results, a stability-preserving discrete implementation of the projection step of the model reduction is outlined.

4.4.1. Stability-preserving discrete implementation

The stability analysis of Sections 4.1–4.2 has assumed that the integrals resulting from the projection of the governing equations onto the reduced basis modes are evaluated exactly in continuous form. This continuous result can be translated to the discrete setting through the use of high-precision numerical quadrature as follows. First, the snapshots and the POD basis modes are cast as a collection of continuous finite elements. It is then possible to construct a numerical quadrature operator that computes exactly (with respect to the finite element representation) all continuous inner products arising from the continuous Galerkin projection of the equations onto the POD modes. Suppose the domain Ω is broken up into n_{el} finite elements Ω_e such that $\cup_{e=1}^{n_{el}} \Omega_e = \Omega$. Suppose each of these elements have nn nodes. Then, the finite element representation of the vector \mathbf{q}' in (27) in each element Ω_e is:

$$\mathbf{q}'_e = \sum_{i=1}^{nn} N_i(\mathbf{x}) \mathbf{q}'_i(\mathbf{x}), \quad \mathbf{x} \in \Omega_e, \quad (66)$$

where the $N_i(\mathbf{x})$ are the finite element shape functions used to represent the solution in each element.

With the representation (66), the modes will necessarily be in the Sobolev space $H^1(\Omega)$, as required for the proof of Theorem 4.1.1.

By the discussion in Section 4.2, it is necessary to compute numerically integrals of the form:

$$(\mathbf{q}^{(1)}, \mathbf{q}^{(2)})_{(\mathbf{H}, \Omega)} = \int_{\Omega} \mathbf{q}^{(1)T} \mathbf{H} \mathbf{q}^{(2)} d\Omega. \quad (67)$$

Suppose, without loss of generality, that the finite element shape functions are chosen to be bilinear, so $nn = 4$. The discrete representations of the vectors $\mathbf{q}^{(1)}$ and $\mathbf{q}^{(2)}$ are denoted by $\mathbf{q}^{h(1)}$ and $\mathbf{q}^{h(2)}$, respectively. The length of these vectors is equal to the number of mesh nodes N times the dimension of the vector, r . Let \mathbf{H}_e^h be the $r \times r$ element inner product matrix, taken to be piecewise constant over each element. Then, the formula for numerical integration of (67) can be written as

$$(\mathbf{q}^{(1)}, \mathbf{q}^{(2)})_{(\mathbf{H}, \Omega)} = \mathbf{q}^{h(1)T} \mathbf{W} \mathbf{q}^{h(2)}, \quad (68)$$

where \mathbf{W} is a sparse block matrix comprised of $N \times N$ blocks of dimension $r \times r$. The $(k, l)^{th}$ block of this matrix given by $w_{kl} \mathbf{I}$, where

$$w_{kl} = \sum_{e=1}^{n_{kl}^{el}} \mathbf{H}_e^h \sum_{j=1}^4 N_{k_e}(\mathbf{x}_{j_e}) N_{l_e}(\mathbf{x}_{j_e}) \omega_{j_e}. \quad (69)$$

Here, the outer sum is over the elements connected to the $k-l$ nodal “edge”; the ω_{j_e} are the integration weights and the \mathbf{x}_{j_e} are the integration points.

A parallel C++ code that reads in the snapshot data written by a high-fidelity code, assembles the necessary finite element representation of the snapshots and computes the numerical quadrature necessary for evaluation of the inner products has been written by the authors. The code, known as *Spirit*, performs all the calculations in parallel using distributed matrix and vector data structures and parallel eigensolvers from the Trilinos project [48], and uses the *libmesh* finite element library [49] to compute element quadratures. The parallelism in *Spirit* allows for large data sets and a relatively large number of POD modes. The *libmesh* finite element library [49] was used to

compute element quadratures. The online time-integration of the ROM system (4) (with the ROM coefficient matrix computed within `Spirit` and written to disk) is then performed using a fourth-order Runge-Kutta scheme in MATLAB. For more information on the `Spirit` code, the reader is referred to [57; 56].

4.4.2. 2D inviscid acoustic pulse example

For the sake of brevity, the proposed model reduction approach is evaluated on only one of the physics sets given in Section 4.3. The test case considered is that of a 2D inviscid acoustic pressure pulse in the following 2D prismatic domain: $\Omega = (-1, 1) \times (-1, 1) \in \mathbb{R}^2$. The governing equations are the linearized compressible Euler equations (Example 3 in Section 4.3). The base flow is uniform, with the following values: $\bar{p} = 101,325$ Pa, $\bar{T} = 300$ K, $\bar{\rho} = \frac{\bar{p}}{RT} = 1.17$ kg/m³, $\bar{u}_1 = \bar{u}_2 = 0.0$ m/s, and $\bar{c} = 348.0$ m/s, where $\bar{c} \equiv \sqrt{\gamma R \bar{T}}$ is the mean speed of sound. The problem is initialized with a pressure pulse in the middle of the domain:

$$\begin{aligned} p'(\mathbf{x}; 0) &= 141.9e^{-10(x^2+y^2)}, \\ \rho'(\mathbf{x}; 0) &= \frac{p'(\mathbf{x}; 0)}{RT}, \\ T'(\mathbf{x}; 0) &= 0, \\ u'_1(\mathbf{x}; 0) &= u'_2(\mathbf{x}; 0) = 0. \end{aligned} \quad (70)$$

In terms of the mean values, the amplitude of the initial pressure pulse (70) is $0.001\bar{p}\bar{c}^2$.

For the problem considered, the high-fidelity fluid simulation data were generated using a Sandia in-house finite volume flow solver known as SIGMA CFD. This code is derived from LESLIE3D [50], a Large Eddy Simulations (LES) flow solver originally developed in the Computational Combustion Laboratory at the Georgia Institute of Technology. For a detailed description of the schemes and models implemented within LESLIE3D, the reader is referred to [51; 52].

As both the high-fidelity code as well as the ROM code are 3D codes, a 2D mesh of the domain Ω is converted to a 3D mesh by extruding the 2D mesh

in the z -direction by one element. The computational grid for this test case is composed of 3362 nodes, cast into 9600 tetrahedral finite elements within the ROM code. A no-penetration (slip wall) boundary condition is imposed on the four sides of the domain in the x and y plane:

$$\mathbf{u}' \cdot \mathbf{n}_{np} = 0 \text{ on } \Gamma_{np}, \quad (71)$$

where $\Gamma_{np} = \{(x, z) \in \Omega : y = 1, -1\} \cup \{(y, z) \in \Omega : x = 1, -1\}$, and \mathbf{n}_{np} denotes the unit normal vector to Γ_{np} . To ensure the solution has no dynamics in the z -direction, the following values of the z -velocity component are specified: $\bar{u}_3 = 0$, $u'_3(\mathbf{x}; 0) = 0$. Symmetry boundary conditions are imposed for $z = \text{constant}$ in the high-fidelity code:

$$\begin{cases} \mathbf{u}' \cdot \mathbf{n}_s = 0, \\ \nabla \mathbf{u}' \cdot \mathbf{n}_s = \mathbf{0}, \\ \nabla p' \cdot \mathbf{n}_s = 0, \\ \nabla \rho' \cdot \mathbf{n}_s = 0, \end{cases} \text{ on } \Gamma_s, \quad (72)$$

where $\Gamma_s = \{(x, y) \in \Omega : z = \text{const}\}$ and \mathbf{n}_s is the unit normal vector to Γ_s . The high-fidelity computational fluid dynamics (CFD) simulation from which the ROM is generated is performed until time $T = 0.01$ seconds. During this simulation, the initial pressure pulse (70) reflected from the walls of the domain a number of times. Snapshots from this simulation were saved every 5×10^{-5} seconds, to yield a total of 200 snapshots. These snapshots were used to construct 20 mode POD bases. Two different procedures were used to generate a fluid ROM for this problem: the POD/Galerkin method with the symmetry inner product (49) with \mathbf{H} given by (64), and the POD/Galerkin method with the classical L^2 inner product. The size of the POD basis was determined using an energy criterion (8) (see Section 2.1): M was selected such that the modes capture 99.9% of the snapshot energy. Since the base flow for this example is uniform, $\mathbf{G} = \mathbf{0}$ and \mathbf{A}_i and \mathbf{K}_{ij} are spatially-constant in (27), meaning an energy-stable ROM is expected to be time-stable and stable in the sense of Lyapunov. Figure 1 shows a time history of the first two ROM modal amplitudes (circles) compared to the projection of the FOM CFD simulation onto the first two POD

modes (solid lines) for the symmetry (a) and L^2 (b) ROMs. Mathematically, this figure compares as a function of time t :

$$x_{M,i}(t) \quad \text{vs.} \quad (\mathbf{q}'_{\text{FOM}}, \boldsymbol{\phi}_i)_{(\mathbf{H}, \Omega)}, \quad (73)$$

for $i = 1, 2$, where \mathbf{q}'_{FOM} is the high-fidelity CFD solution from which the ROMs were constructed. The reader may observe reasonable agreement between the symmetry ROM and the full simulation (Figure 1(a)) for the time interval considered. In contrast, agreement between the L^2 ROM and the full simulation is reasonable only until approximately $t = 0.005$ seconds (Figure 1(b)). The oscillations in the L^2 ROM modal amplitudes observed for $t > 0.008$ seconds suggest the presence of an instability in the L^2 ROM. If the modal amplitudes $x_{M,i}(t)$ are plotted up to a longer time horizon (Figure 2), the instability in the L^2 ROM is apparent.

Figures 3–4 compare the FOM pressure field (a) with the field reconstructed from the symmetry (b) and L^2 (c) ROM solutions at times $t = 4.5 \times 10^{-4}$ and 7.95×10^{-3} seconds. At time $t = 4.5 \times 10^{-4}$ seconds, both the symmetry and L^2 ROM solutions are in good agreement with the high-fidelity solution (Figure 3). At the later time, 7.95×10^{-3} seconds, there is a good qualitative agreement between the high-fidelity solution and the symmetry ROM solution (Figure 4(a), (b)). The same cannot be said of the L^2 ROM solution, however. It is apparent from Figure 4(c) that the L^2 ROM solution has blown up by $t = 7.95 \times 10^{-3}$ seconds, which confirms the instability of the 20 mode L^2 ROM suggested in Figures 1–2.

5. Stable model reduction for LTI systems via discrete projection

In Section 4, a method for constructing energy-stable ROMs via continuous projection of a linear system of PDEs was presented. The discussion in Section 4 motivates the following question: can the energy inner product be determined in a black-box fashion for any given full order model system? It is shown in the present section that there is a discrete

counterpart of the symmetry inner product, first derived by Rowley *et al.* [24] and termed the “Lyapunov inner product” herein. Although the Lyapunov inner product has appeared in several publications [24; 1; 30], to the authors’ knowledge, a numerical study of the properties and performance of POD ROMs constructed in the Lyapunov inner product is lacking from the literature at the present time, and one of the contributions of this work.

5.1. Stability-preserving Lyapunov inner product and Petrov-Galerkin connection

Suppose the LTI system (4) is stable in the sense of Lyapunov, i.e., all eigenvalues of the matrix \mathbf{A} have non-positive real parts (Corollary 3.4.2). Since \mathbf{A} is stable, there exists a Lyapunov function for

$$\dot{\mathbf{x}}_N(t) = \mathbf{A}\mathbf{x}_N(t). \quad (74)$$

In particular,

$$V(\mathbf{x}_N) = \mathbf{x}_N^T \mathbf{P} \mathbf{x}_N, \quad (75)$$

is a Lyapunov function for (74), where \mathbf{P} is the solution of the following Lyapunov equation:

$$\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} = -\mathbf{Q}. \quad (76)$$

Here, \mathbf{Q} is some positive-definite matrix [15]. A positive definite solution \mathbf{P} to (76) exists provided \mathbf{A} is stable. Moreover, if \mathbf{Q} is symmetric, \mathbf{P} is symmetric as well. Given \mathbf{A} and \mathbf{Q} , a solution to the Lyapunov equation (76) can be obtained, for instance, using the `lyap` function in the MATLAB control toolbox [43]:

$$\mathbf{P} = \text{lyap}(\mathbf{A}', \mathbf{Q}, [], \text{speye}(\mathbf{N}, \mathbf{N})).$$

Assume the system (74) is stable and a positive-definite symmetric \mathbf{P} has been computed from (76). Since \mathbf{P} is symmetric positive-definite, the following

$$\left(\mathbf{x}_N^{(1)}, \mathbf{x}_N^{(2)} \right)_{\mathbf{P}} \equiv \mathbf{x}_N^{(1)T} \mathbf{P} \mathbf{x}_N^{(2)}, \quad (77)$$

defines an inner product. Let $\boldsymbol{\Phi}_M$ be a reduced basis of size M , so that

$$\mathbf{x}_N(t) \approx \boldsymbol{\Phi}_M \mathbf{x}_M(t), \quad (78)$$

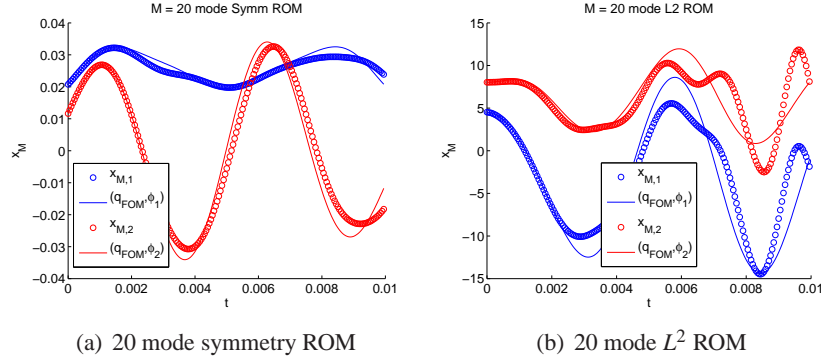


Figure 1: Time history of modal amplitudes for inviscid pressure pulse problem

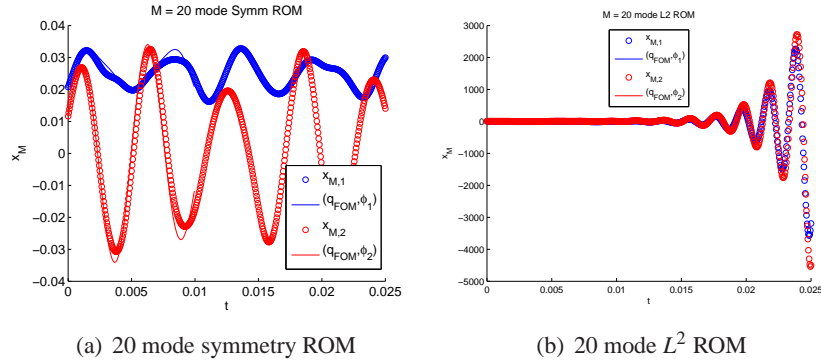


Figure 2: Time history of modal amplitudes for inviscid pressure pulse problem for longer time horizon

where $\mathbf{x}_M(t)$ denotes the ROM solution. Theorem 5.1.1 (summarized here from Section 2.3 of [24] to keep this work self-contained) shows that (77) is the energy inner product for this system.

Theorem 5.1.1 (from Section 2.3 of [24]): Assume the linear full order system (74) is stable. Suppose a ROM for (74) is constructed via a Galerkin projection in the $(\cdot, \cdot)_{\mathbf{P}}$ inner product (77), to yield the following reduced linear system:

$$\dot{\mathbf{x}}_M = \Phi_M^T \mathbf{P} \mathbf{A} \Phi_M \mathbf{x}_M, \quad (79)$$

where it has been assumed that the basis Φ_M has been constructed to be orthonormal in the $(\cdot, \cdot)_{\mathbf{P}}$ inner product, i.e., $\Phi_M^T \mathbf{P} \Phi_M = \mathbf{I}_M$ where \mathbf{I}_M denotes the $M \times M$ identity matrix. Then, the ROM (79) is energy-stable, time-stable and stable in the sense of Lyapunov.

Proof. It is shown that the energy $E_M \equiv \frac{1}{2} \|\mathbf{x}_M\|_2^2$

of the ROM system (79) is non-increasing:

$$\begin{aligned} \frac{dE_M}{dt} &= \frac{1}{2} \frac{d}{dt} (\mathbf{x}_M, \mathbf{x}_M)_2 \\ &= \mathbf{x}_M^T \dot{\mathbf{x}}_M \\ &= \mathbf{x}_M^T \Phi_M^T \mathbf{P} \mathbf{A} \Phi_M \mathbf{x}_M \\ &= \mathbf{x}_M^T \Phi_M^T \left(\frac{1}{2} \mathbf{P} \mathbf{A} + \frac{1}{2} \mathbf{P}^T \mathbf{A} \right) \Phi_M \mathbf{x}_M \quad (80) \\ &= \mathbf{x}_M^T \Phi_M^T \left(\frac{1}{2} \mathbf{P} \mathbf{A} + \frac{1}{2} \mathbf{A}^T \mathbf{P} \right) \Phi_M \mathbf{x}_M \\ &= -\frac{1}{2} \mathbf{x}_M^T \Phi_M^T \mathbf{Q} \Phi_M \mathbf{x}_M \\ &< 0, \end{aligned}$$

since $\mathbf{Q} > \mathbf{0}$. It follows that (79) is time-stable, stable in the sense of Lyapunov and energy-stable (Section 3).

□

The Lyapunov inner product (77) is a discrete counterpart of the continuous symmetry inner product (49). This inner product can be employed to construct stable Galerkin ROMs for (4) using discrete projection. An interesting question that

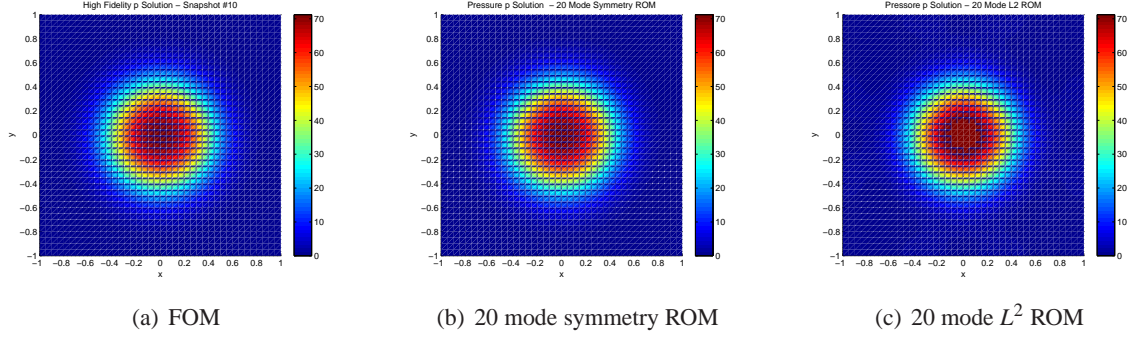


Figure 3: Pressure field at time $t = 4.5 \times 10^{-4}$ seconds

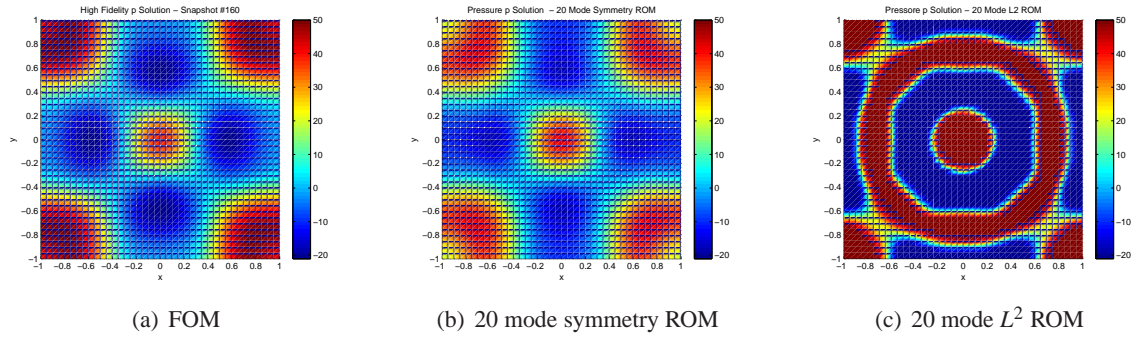


Figure 4: Pressure field at time $t = 7.95 \times 10^{-3}$ seconds

arises is whether the matrix \mathbf{P} defining the Lyapunov inner product (77) is related in some way to the matrix \mathbf{W} (68) that is used to perform the continuous projection in the symmetry inner product. In general, the answer is no. In particular, \mathbf{W} is by construction a sparse matrix (Figure 5(a)), whereas \mathbf{P} may be dense even if \mathbf{A} is sparse. This is clear from Figures 5(b) and (c), which show (respectively) the sparsity pattern of a sample \mathbf{A} matrix⁵, and its corresponding \mathbf{P} matrix.

One downside of the Lyapunov inner product is that the matrix \mathbf{P} which defines this inner product is admittedly expensive to compute: the cost of solving the Lyapunov equation (76) requires $\mathcal{O}(N^3)$ operations. As a consequence, the Lyapunov inner product has the same downside as another model reduction approach with an *a priori* stability guar-

antee, namely balanced truncation [29; 10]: it may not be practical to compute the matrix \mathbf{P} defining the Lyapunov inner product for very large systems. It is worthwhile to note that computing \mathbf{P} (76) is less computationally intensive than reducing a system using balanced truncation, which requires the solution of *two* Lyapunov equations for the so-called observability and reachability Gramians *and* the factorizations of these Gramians [29; 10] (see Appendix A.2). The computational cost of calculating the weighting matrix that defines the Lyapunov inner product relative to the computational cost of reducing a system using balanced truncation is studied numerically in Section 5.2. Note that it can be shown that the balanced truncation algorithm may be viewed as a projection algorithm in a special Lyapunov inner product [24]. A proof uncovering this connection is given in Appendix A.3.

As observed earlier for the symmetry inner product, it is clear from (79) that the Galerkin projection

⁵The \mathbf{A} matrix whose sparsity pattern is shown in Figure 5(b) is the “PDE example” in the SLICOT model reduction benchmark repository [41].

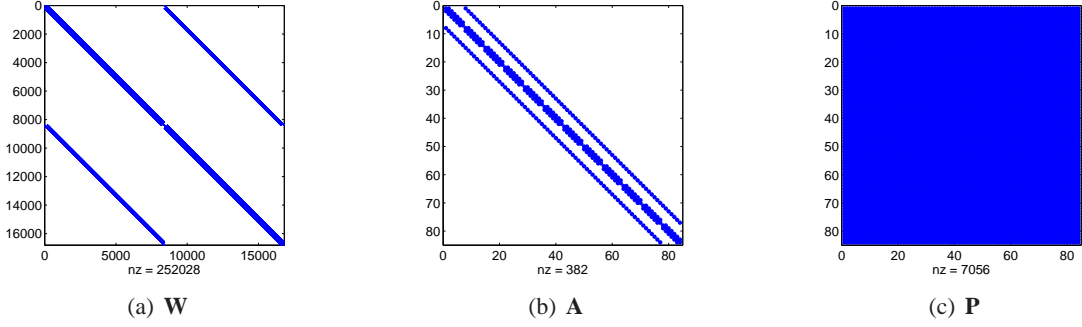


Figure 5: Sparsity structure of representative \mathbf{P} matrix for a given sparse \mathbf{A} matrix compared to sparsity structure of representative \mathbf{W} matrix

of the system (74) in the Lyapunov inner product (77) can be viewed as a Petrov-Galerkin projection of this system in the regular L^2 inner product, with the reduced test basis given by $\Psi_M = \mathbf{P}\Phi_M$, where Φ_M is the reduced trial basis.

5.2. Numerical experiments

The stability-preserving model reduction approach based on discrete projection presented in Section 5.1 is now evaluated on a problem involving a model of an electrostatically actuated beam. For this example, the error in the ROM output relative to the full order model output, defined by

$$\mathcal{E}_{rel}^{eo} = \frac{\sum_{i=1}^{K_{max}} |\mathbf{y}_{QN}(t_i) - \mathbf{y}_{QM}(t_i)|}{\sum_{i=1}^{K_{max}} |\mathbf{y}_{QN}(t_i)|}, \quad (81)$$

is computed and reported. Here the symbol K_{max} denotes the integer such that $T_{max} = K_{max}dt_{snap}$, where T_{max} is the maximum time until which the ROM is run. The notation $|\cdot|$ in (81) denotes the absolute value, which evaluates to a scalar for the numerical example considered, as it has one output ($Q = 1$).

5.2.1. Electrostatically actuated beam example

The numerical example considered is that of an electrostatically actuated beam. One application for this model is analysis of microelectromechanical systems (MEMS) devices, such as electromechanical radio frequency (RF) filters [44]. Given a

simple enough shape, these devices can be modeled as 1D beams embedded in two or three dimensional space. It is assumed that the beam deflection is small, so that geometric nonlinearities can be neglected. The resulting linear PDEs are discretized using the finite element method following the approach presented in [45; 44] to yield a ROM LTI system of the form (4). The matrices \mathbf{A} and \mathbf{B} in (4) are downloaded from the Oberwolfach model reduction benchmark collection [46]. These global matrices are then disassembled into their local counterparts, and reassembled to yield a discretization of any desired size. In the full order model for which results are reported here, the FOM has $N = 10,000$ degrees of freedom. It is verified that the full order system is stable: the maximum real part of the eigenvalues of \mathbf{A} is -0.0016 .

To generate the snapshots from which POD bases are constructed, the full order model (4) is solved using a backward Euler time integration scheme with an initial condition of $\mathbf{x}_N(0) = \mathbf{0}$ and an input corresponding to a periodic on/off switching, i.e.,

$$\mathbf{u}_P(t) = \begin{cases} 0.005 < t < 0.01, 0.015 < t < 0.02, \\ 1, & 0.03 < t < 0.035, \\ 0, & \text{otherwise} \end{cases} \quad (82)$$

A total of $K_{max} = 1000$ snapshots are collected, every $dt_{snap} = 5 \times 10^{-5}$ seconds, until time $t = 0.05$ seconds. From these snapshots, 5, 10, 20 and 30 mode ROMs are constructed using POD in the L^2 inner product, and POD in the Lyapunov inner product. In solving the Lyapunov equation (76)

for the Lyapunov inner product weighting matrix \mathbf{P} , the matrix \mathbf{Q} is taken to be the $N \times N$ identity matrix. The system (4) is reduced also using balanced truncation.

The first step is to study the stability of each ROM. Table 1 reports the maximum real part of the ROM system matrices \mathbf{A}_M for each M considered. It is found that the L^2 ROM is unstable for each M , and becomes more unstable with increasing M . In contrast, the balanced truncation and POD Lyapunov inner product ROMs are stable for all M considered, as expected.

Next, the accuracy of each ROM is examined. Table 2 summarizes the errors (81) in the ROM solutions relative to the full order model solution for three runs of different lengths. An entry of ‘–’ in the table indicates that the error overflowed due to an instability in the ROM.

The objective of the first run ($K_{max} = 1000$) is to study how well the POD ROMs can reproduce the snapshots from which they were constructed, and to compare these ROMs’ performance with the performance of ROMs constructed using balanced truncation. The reader can observe that the POD ROM constructed in the Lyapunov inner product is the most accurate. The POD L^2 ROM is both unstable as well as inaccurate (Figure 6(a)).

The second two runs ($K_{max} = 2000$ and $K_{max} = 5000$) are aimed to study the predictive capabilities of the ROMs for long-time simulations. The full order model is run until times 0.1 and 2.5 seconds respectively. As before, only snapshots up to time $t = 0.05$ seconds are used to construct the POD bases for the ROMs. In addition to the signal (82), the following inputs are applied in both the full order model and the ROM:

$$\mathbf{u}_P(t) = \begin{cases} 0.055 < t < 0.06, 0.065 < t < 0.07, \\ 0.08 < t < 0.085, 0.105 < t < 0.11, \\ 0.115 < t < 0.12, 0.13 < t < 0.135, \\ 0.205 < t < 0.21, 0.215 < t < 0.22, \\ 1, & 0.23 < t < 0.235, \\ 0, & \text{otherwise.} \end{cases} \quad (83)$$

The reader may observe by examining Table 2 and Figure 6 that the balanced truncation ROMs are in

general the most accurate. The POD ROMs constructed in the Lyapunov inner product nonetheless produce reasonable results (Figures 6(b)-(c)) and appear to be converging to the full order model solution with M -refinement (Table 2). The POD L^2 ROM result is not shown in Figures 6(b)-(c), as the solution produced by this ROM blows up around time $t = 0.02$ seconds.

Lastly, the level of computational resources required for computing the Lyapunov inner product and the level of computational resources required for performing model reduction via balanced truncation [29; 10] are compared. Table 3 gives the CPU times for the sum of the following operations in the balanced truncation [29; 10] algorithm as a function of N , the problem size: calculation of the observability Gramian, calculation of the controllability Gramian, and calculation of the balancing transformation (Appendix A.2). All computations are performed in serial using MATLAB’s linear algebra capabilities and MATLAB’s control toolbox [43], on a Linux workstation with 6 Intel Xeon 2.93 GHz CPUs. Both methods exhibit $O(N^3)$ scaling. Although the Lyapunov inner product computation is costly, as it requires the solution of a Lyapunov equation, it completes in 2-3 times less CPU time than the balanced truncation algorithm. This is because balanced truncation requires the solution of *two* Lyapunov equations for the observability and reachability Gramians, as well as the Cholesky and eigenvalue factorizations of these Gramians.

6. Summary and conclusions

The energy-stability preserving model reduction approach developed specifically for the equations of linearized compressible inviscid flow in [6; 7] is generalized: for ROMs constructed using the continuous projection approach, it is shown that a transformation of a generic PDE system of the hyperbolic or incompletely parabolic type leads to a stable formulation of the Galerkin ROM for this system. It is then shown that, for many linear PDE systems, the said transformation is induced by a

Table 1: Maximum real part of eigenvalues of ROM system matrix \mathbf{A}_M for electrostatically actuated beam problem as a function of basis size M

Basis Size (M)	Balanced Truncation	POD L^2	POD Lyapunov \mathbf{P}
5	-2.97×10^{-6}	4.51×10^1	-3.15×10^1
10	-3.95×10^{-6}	1.39×10^2	-1.89×10^1
20	-2.78×10^{-6}	1.68×10^3	-7.48
30	-3.22×10^{-6}	3.40×10^3	-4.37

Table 2: Relative errors (81) \mathcal{E}_{rel}^o in ROM output for electrostatically actuated beam problem

K_{max}	Method	M			
		5	10	20	30
1000	BT	6.29×10^{-2}	4.51×10^{-3}	6.93×10^{-5}	3.60×10^{-6}
	POD L^2	8.56×10^{-1}	6.62	—	—
	POD Lyapunov \mathbf{P}	2.05×10^{-3}	6.23×10^{-5}	2.09×10^{-8}	1.35×10^{-8}
2000	BT	5.84×10^{-2}	4.47×10^{-3}	6.29×10^{-5}	3.17×10^{-6}
	POD L^2	7.76	4.26×10^3	—	—
	POD Lyapunov \mathbf{P}	3.62×10^{-2}	1.12×10^{-2}	3.47×10^{-4}	4.13×10^{-5}
5000	BT	7.36×10^{-2}	4.77×10^{-3}	5.48×10^{-5}	2.77×10^{-6}
	POD L^2	4.40×10^3	—	—	—
	POD Lyapunov \mathbf{P}	1.80×10^{-1}	1.09×10^{-1}	2.03×10^{-2}	6.09×10^{-3}

Table 3: CPU Times (in seconds) for balanced truncation vs. Lyapunov inner product computations

Method	N			
	1250	2500	5000	10,000
Lyapunov Inner Product	5.08×10^1	4.60×10^2	4.02×10^3	6.09×10^4
Balanced Truncation	1.09×10^2	1.10×10^3	1.04×10^4	1.24×10^5

special inner product, referred to as the “symmetry inner product”. If the Galerkin projection step of the model reduction procedure is performed in this inner product, the resulting ROM is guaranteed to satisfy certain stability bounds regardless of the reduced basis employed. It is demonstrated that a discrete counterpart of the symmetry inner product is the weighted L^2 inner product obtained by solving a Lyapunov equation, derived in [24] by Rowley *et al.* For completeness, this inner product, referred to as the “Lyapunov inner product”, is re-derived herein, and it is shown using the energy method that this inner product gives rise to stable ROMs constructed via discrete projection. The performance of POD ROMs constructed using the symmetry and Lyapunov inner products are assessed on several numerical examples for which POD ROMs constructed in the L^2 inner product

manifest instabilities.

The key properties of the symmetry inner product and Lyapunov inner product are summarized in Table 4. Both inner products are weighted L^2 inner products and have the same origin: they are induced by the Lyapunov function for the governing system of equations. The symmetry inner product is a continuous inner product derived for a specific PDE system of the form (27). Projection in this inner product requires access to the governing PDEs, which gives rise to a projection algorithm that is embedded. The Lyapunov inner product is discrete, on the other hand, and operates on an LTI system of the form (4) arising from the discretization of a PDE of the form (1) in space using some numerical scheme, e.g., the finite element method. Projection in the Lyapunov inner product is there-

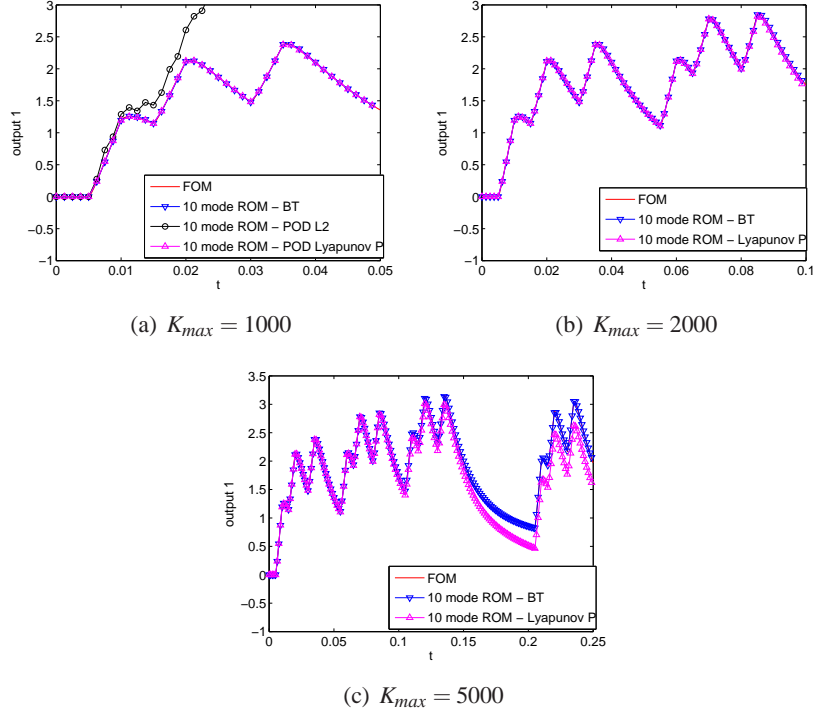


Figure 6: $y_{QM}(t)$ for $M = 10$ ROMs (FOM = full order model) for electrostatically actuated beam problem

fore a black-box algorithm, as only the \mathbf{A} , \mathbf{B} and \mathbf{C} matrices in (4) are needed; in particular, access to the governing equations is *not* required. The symmetric positive definite matrix that defines the Lyapunov inner product can also be computed numerically in a black-box fashion by solving a Lyapunov equation. The existence of a solution to this Lyapunov equation is certain only if the full order system (4) is stable; hence the Lyapunov inner product is not defined for unstable systems. In contrast, the symmetry inner product *is* defined for unstable systems. In this case, a ROM constructed in this inner product will be energy-stable, by construction. However, it will not be time-stable, i.e., it may produce (physical) solutions that are unbounded as $t \rightarrow \infty$. The discussion above may lead the reader to prefer the Lyapunov inner product to the symmetry inner product, as the former inner product can be computed in a black-box fashion for any stable linear system, and can be used to build a ROM for this system without accessing the PDEs. One of the biggest drawbacks of the Lyapunov inner prod-

uct projection approach involves its large computational cost. To solve numerically the Lyapunov equation that defines this inner product, $\mathcal{O}(N^3)$ operations are required. Moreover, since the matrix that defines the Lyapunov inner product is typically dense (in contrast to the matrix defining the symmetry inner product, which is sparse), at least $\mathcal{O}(N^2)$ storage is required [11]. As a result, creating ROMs using the Lyapunov inner product may not be practical for systems of very large size. The Lyapunov inner product may nonetheless be preferable to balanced truncation, which requires the solution of two Lyapunov equations, and the storage of two Gramians, in addition to Cholesky and eigenvalue factorization of these Gramians. For large-scale unsteady problems, the symmetry inner product combined with the continuous projection approach is recommended by the authors, despite its more involved implementation.

It is worthwhile to note that there exist in the literature many efficient, low storage algorithms

Table 4: Comparison of symmetry inner product and Lyapunov inner product

Symmetry Inner Product (49)	Lyapunov Inner Product (77)
Continuous	Discrete
For linear PDE system of the form $\dot{\mathbf{q}}' + \mathbf{A}_i \frac{\partial \mathbf{q}'}{\partial x_i} + \mathbf{K}_{ij} \frac{\partial^2 \mathbf{q}'}{\partial x_i \partial x_j} + \mathbf{G} \mathbf{q}' = \mathbf{f}$	For linear ODE system of the form $\dot{\mathbf{x}}_N = \mathbf{A} \mathbf{x}_N$
Defined for unstable systems but time-stability of ROM is not guaranteed	Undefined for unstable systems
Induced by Lyapunov function for the system	Induced by Lyapunov function for the system
Equation specific	Black-box
Derived analytically in closed form	Computed numerically by solving a Lyapunov equation
Sparse	Dense

to compute approximate solutions to large-scale Lyapunov equations, like the equation defining the Lyapunov inner product, e.g., the work of Li [58; 59], Benner [60] and Simoncini [61]. These approximate Lyapunov solvers can be used to perform approximate balanced truncation model reduction, and can, in a similar fashion, be used to compute an approximation of the Lyapunov inner product. With these approximate methods, however, the theoretical *a priori* stability guarantee shown herein for ROMs constructed in the Lyapunov inner product is lost, in general. Likely, ROMs constructed in approximate Lyapunov inner products will in practice have better numerical stability properties than ROMs constructed in the L^2 inner product. A numerical study of the performance of such ROMs is a worthwhile future research endeavor that may be the subject of future work. It is beyond the scope of the present article, which focuses on inner products that give rise to Galerkin formulations with an *a priori* stability guarantee.

Appendix

A.1. Gronwall's Lemma

Gronwall's lemma (also known as Gronwall's inequality) allows one to bound a function that is

known to satisfy a certain differential or integral inequality by the solution of the corresponding differential or integral equation [54]. The differential form of this inequality is used herein:

$$\dot{\mathbf{x}}(t) \leq \beta(t)\mathbf{x}(t) \Rightarrow \mathbf{x}(T) \leq \mathbf{x}(0)e^{\int_0^T \beta(s)ds} \quad (84)$$

for $\beta \in L^2, t \geq 0, 0 \leq T \leq t$.

A.2. Balanced truncation algorithm for model reduction

The balanced truncation algorithm, first introduced by Moore [29], assumes a semi-discrete full order model of the form (4). The linear system (4) is first transformed into a balanced form that isolates observable and reachable (or controllable) modes. This is achieved by simultaneously diagonalizing the reachability (or controllability) and observability Gramians. The reachability (or controllability) Gramian (Chapter 30 of [17])

$$\mathbf{P} \equiv \int_0^\infty e^{\mathbf{A}t} \mathbf{B} \mathbf{B}^T e^{\mathbf{A}^T t} dt, \quad (85)$$

is the unique symmetric (at least) positive semi-definite solution of the Lyapunov equation

$$\mathbf{A} \mathbf{P} + \mathbf{P} \mathbf{A}^T + \mathbf{B} \mathbf{B}^T = \mathbf{0}. \quad (86)$$

The observability Gramian (Chapter 30 of [17])

$$\mathbf{Q} \equiv \int_0^\infty e^{\mathbf{A}^T t} \mathbf{C}^T \mathbf{C} e^{\mathbf{A}t} dt, \quad (87)$$

is the unique symmetric (at least) positive semi-definite solution of the Lyapunov equation

$$\mathbf{A}^T \mathbf{Q} + \mathbf{Q} \mathbf{A} + \mathbf{C}^T \mathbf{C} = \mathbf{0}. \quad (88)$$

It will be assumed herein that the matrix \mathbf{A} defining the full order system (4) is stable, i.e., it has no eigenvalues with a positive real part. It will also be assumed (\mathbf{A}, \mathbf{C}) is observable and (\mathbf{A}, \mathbf{B}) is reachable (controllable). If this is true, the Lyapunov equations (86) and (88) will have positive definite solutions \mathbf{P} and \mathbf{Q} respectively (Chapter 6 of [18]). For a discussion of balanced truncation applied to unstable systems, the reader is referred to [22].

The balanced truncation algorithm is summarized below for the specific case of real system matrices⁶ \mathbf{A} , \mathbf{B} and \mathbf{C} . First, the reachability Gramian \mathbf{P} is obtained by solving the Lyapunov equation (86). Next, the observability Gramian \mathbf{Q} is obtained by solving the Lyapunov equation (88). The Cholesky factorization of \mathbf{P} is computed,

$$\mathbf{P} = \mathbf{U} \mathbf{U}^T. \quad (89)$$

followed by an eigenvalue decomposition of $\mathbf{U}^T \mathbf{Q} \mathbf{U}$:

$$\mathbf{U}^T \mathbf{Q} \mathbf{U} = \mathbf{K} \mathbf{\Sigma}^2 \mathbf{K}^T. \quad (90)$$

The balancing transformation matrices:

$$\mathbf{T}_{bal} = \mathbf{\Sigma}^{1/2} \mathbf{K}^T \mathbf{U}^{-1}, \quad \mathbf{T}_{bal}^{-1} = \mathbf{U} \mathbf{K} \mathbf{\Sigma}^{-1/2}, \quad (91)$$

can now be computed⁷, where the entries of $\mathbf{\Sigma}$ are in decreasing order. The change of variables $\tilde{\mathbf{x}}_N(t) = \mathbf{T}_{bal} \mathbf{x}_N(t)$ is applied to the full-order LTI system (4) to yield:

$$\begin{aligned} \dot{\tilde{\mathbf{x}}}_N(t) &= \mathbf{T}_{bal} \mathbf{A} \mathbf{T}_{bal}^{-1} \tilde{\mathbf{x}}_N(t) + \mathbf{T}_{bal} \mathbf{B} \mathbf{u}_P(t), \\ \mathbf{y}_{QN}(t) &= \mathbf{C} \mathbf{T}_{bal}^{-1} \tilde{\mathbf{x}}_N(t). \end{aligned} \quad (92)$$

⁶In the case these matrices are complex, the transpose operation T in the algorithm (and all analysis of this algorithm) should be replaced with a Hermitian transpose H .

⁷In practice, the transformation matrices (91) are typically computed as $\mathbf{T}_{bal} = \mathbf{V}^T \mathbf{Z}^T$, and $\mathbf{T}_{bal}^{-1} = \mathbf{U} \mathbf{W}$, where \mathbf{Z} is the Cholesky factor of the observability Gramian ($\mathbf{Q} = \mathbf{Z} \mathbf{Z}^T$), and \mathbf{W} is the left singular vector of $\mathbf{U}^T \mathbf{Z}$ ($\mathbf{U}^T \mathbf{Z} = \mathbf{W} \mathbf{\Sigma} \mathbf{V}^T$). This is due to numerical stability issues that could arise in computing $\mathbf{\Sigma}^{-1/2}$ in (91).

Next, the matrices $\tilde{\mathbf{A}} \equiv \mathbf{T}_{bal} \mathbf{A} \mathbf{T}_{bal}^{-1}$, $\tilde{\mathbf{B}} \equiv \mathbf{T}_{bal} \mathbf{B}$, $\tilde{\mathbf{C}} \equiv \mathbf{C} \mathbf{T}_{bal}^{-1}$ are partitioned as follows:

$$\begin{aligned} \tilde{\mathbf{A}} &= \left(\begin{array}{c|c} \tilde{\mathbf{A}}_{11} & \tilde{\mathbf{A}}_{12} \\ \hline \tilde{\mathbf{A}}_{21} & \tilde{\mathbf{A}}_{22} \end{array} \right), & \tilde{\mathbf{B}} &= \left(\begin{array}{c} \tilde{\mathbf{B}}_1 \\ \tilde{\mathbf{B}}_2 \end{array} \right), \\ \tilde{\mathbf{C}} &= \left(\begin{array}{c|c} \tilde{\mathbf{C}}_1 & \tilde{\mathbf{C}}_2 \end{array} \right). \end{aligned} \quad (93)$$

Here, the blocks with subscript 1 correspond to the most observable and reachable states, and blocks with subscript 2 correspond to the least observable and reachable states. Finally, the reduced system for a ROM of size M is given by:

$$\begin{aligned} \dot{\mathbf{x}}_M(t) &= \mathbf{A}_M \mathbf{x}_M(t) + \mathbf{B}_M \mathbf{u}_P(t), \\ \mathbf{y}_{QM}(t) &= \mathbf{C}_M \mathbf{x}_M(t), \end{aligned} \quad (94)$$

where $\mathbf{A}_M = \tilde{\mathbf{A}}_{11}$, $\mathbf{B}_M = \tilde{\mathbf{B}}_1$, $\mathbf{C}_M = \tilde{\mathbf{C}}_1$. The left and right reduced bases are given respectively by:

$$\mathbf{\Psi}_M = \mathbf{T}_{bal}^T(:, 1:M), \quad \mathbf{\Phi}_M = \mathbf{S}_{bal}(:, 1:M), \quad (95)$$

where $\mathbf{S}_{bal} \equiv \mathbf{T}_{bal}^{-1}$.

In effect, balanced truncation is a method for computing the test and trial bases $\mathbf{\Psi}_M$ and $\mathbf{\Phi}_M$ in (16). Given the test and trial bases defined in (95), the ROM system matrices (94) can be obtained from the formulas (17). The entries of the diagonal matrix $\mathbf{\Sigma}$ in (91) are known as the Hankel singular values of the system (4). Assuming a ROM of size M has been constructed using balanced truncation, the following error bound on the output can be shown [31]:

$$\|\mathbf{y}_{QN}(t) - \mathbf{y}_{QM}(t)\|_2 \leq 2 \sum_{i=M+1}^N \sigma_i \|\mathbf{u}_P(t)\|_2. \quad (96)$$

Generally, balanced truncation is viewed as the “gold standard” in model reduction. Although it is not optimal in the sense that there may be other ROMs with smaller error norms, the approach has *a priori* error bounds that are close to the lowest bounds achievable by any reduced order model [23]. Unfortunately, balanced truncation becomes computationally intractable for systems of very large dimension (e.g., of size $N \geq 10,000$), and hence is not practical for many systems of physical

interest [24]. This is due to the high computational cost of solving the Lyapunov equations (86) and (88) for the reachability and observability Gramians ($\mathcal{O}(N^3)$ operations). The storage requirements of balanced truncation can be prohibitive as well. Even efficient iterative schemes developed for large sparse Lyapunov equations compute the solution to (86) and (88) in dense form, and hence require $\mathcal{O}(N^2)$ storage [11]. Unlike POD, balanced truncation delivers ROMs that preserve stability of a stable system (4) [29], however.

Note that there exist methods for performing model reduction via an approximate balanced truncation, namely balanced POD [23], as well as low-storage solvers for calculating efficiently approximate solutions of large-scale Lyapunov equations like those arising in balanced truncation [58; 59; 60; 61]. Unlike balanced truncation, these algorithms, by construction, *can* be applied to large-scale problems. Although these methods have been found to exhibit better numerical stability properties than the POD/Galerkin method to model reduction, they lack in general the provable *a priori* stability guarantee of balanced truncation.

A.3. Lyapunov inner product associated with balanced truncation

In comparing the steps of the balanced truncation algorithm with the discussion in Section 5.1, the reader may observe some similarities. In particular, both algorithms require the solution of a Lyapunov equation for a Gramian used to transform and reduce the system. Here, this connection is investigated further. In particular, it is shown that the balanced truncation algorithm (Appendix A.2) may be viewed as a projection algorithm in a special Lyapunov inner product.

Suppose the stable LTI system (4) has been reduced using the balanced truncation model reduction algorithm summarized in Appendix A.2. In order to uncover the inner product associated with balanced truncation, several transformations are required.

The first step is to substitute (91) into (95). Then,

the following expressions for the left and right bases are obtained:

$$\Psi_M^T = \mathbf{T}_{bal}(1:M, :) = \Sigma^{1/2}(1:M, :)\mathbf{K}^T\mathbf{U}^{-1}, \quad (97)$$

$$\Phi_M = \mathbf{S}_{bal}(:, 1:M) = \mathbf{U}\mathbf{K}\Sigma^{-1/2}(:, 1:M). \quad (98)$$

Remark that (97) and (98) satisfy the following identity:

$$\Sigma^{-1}(1:M, 1:M)\Psi_M^T\mathbf{P} = \Phi_M^T, \quad (99)$$

where \mathbf{P} is the reachability Gramian (89). It follows that the ROM system matrices in (94) are:

$$\mathbf{A}_M = \Psi_M^T\mathbf{A}\Phi = \Psi_M^T\mathbf{A}\mathbf{P}^T\Psi_M\Sigma^{-1}(1:M, 1:M), \quad (100)$$

$$\mathbf{B}_M = \Psi_M^T\mathbf{B}, \quad (101)$$

$$\mathbf{C}_M = \mathbf{C}\Phi = \mathbf{C}\mathbf{P}^T\Psi_M\Sigma^{-1}(1:M, 1:M). \quad (102)$$

Defining

$$\mathbf{z}_M(t) \equiv \Sigma^{-1/2}(1:M, 1:M)\mathbf{x}_M(t), \quad (103)$$

and employing the symmetry property of the reachability Gramian ($\mathbf{P} = \mathbf{P}^T$), (94) becomes:

$$\begin{aligned} \dot{\mathbf{z}}_M(t) &= \hat{\Psi}_M^T\mathbf{A}\mathbf{P}\hat{\Psi}_M\mathbf{z}_M(t) + \hat{\Psi}_M^T\mathbf{B}\mathbf{u}_P(t), \\ \mathbf{y}_{QM}(t) &= \mathbf{C}\mathbf{P}\hat{\Psi}_M\mathbf{z}_M(t), \end{aligned} \quad (104)$$

where

$$\hat{\Psi}_M \equiv \Psi_M\Sigma^{-1/2}(1:M, 1:M). \quad (105)$$

It is clear that (104) defines a projection of the original LTI system (4) in an L^2 inner product weighted by the reachability Gramian matrix \mathbf{P} . This matrix defines a true inner product in the case when \mathbf{P} is symmetric positive-definite, which will hold if (\mathbf{A}, \mathbf{B}) is reachable (controllable)⁸.

A property of balanced truncation is that it preserves stability when applied to stable systems [10] (Appendix A.2). This result can be proven using the energy method. The proof is analogous to the proof of Theorem 5.1.1.

⁸Reachability (a.k.a. controllability) is a standard concept in control theory. The author is referred to [15] for a detailed discussion of reachability (controllability).

A.4. Weak and penalty method implementations of ROM boundary conditions

Consider a linear PDE system of the form (35) in an open bounded domain Ω with boundary $\partial\Omega$. Assume, without loss of generality, that a transient, Dirichlet boundary condition is imposed on $\partial\Omega$:

$$\mathbf{v}'(\mathbf{x}, t) = \mathbf{g}(\mathbf{x}, t), \quad \text{on } \partial\Omega, \quad (106)$$

for some given, smooth function of boundary data $\mathbf{g}(\mathbf{x}, t)$. Detailed below are two ways to enforce the boundary condition (106) in a ROM constructed via continuous Galerkin projection.

It is often possible to prove stability of a ROM constructed using continuous Galerkin projection with boundary conditions implemented using the weak or penalty methods. For an example of how to do this in the context of the linearized compressible Euler equations with non-reflecting far-field and acoustically-reflecting solid wall boundary conditions, the reader is referred to [7].

A.4.1. Weak implementation of boundary conditions

Projecting (35) onto the k^{th} reduced basis mode, ϕ_k for \mathbf{v}'_M , after applying the identities in (40), (43) and (45), and integrating by parts gives:

$$\begin{aligned} \int_{\Omega} \phi_k^T \dot{\mathbf{v}}'_M d\Omega &= -\frac{1}{2} \int_{\partial\Omega} \phi_k^T \left(\mathbf{A}_i^S + \frac{\partial \mathbf{K}_{ji}^S}{\partial x_j} \right) n_i \mathbf{v}'_M d\Gamma \\ &+ \int_{\partial\Omega} \phi_k^T \mathbf{K}_{ij}^S \frac{\partial \mathbf{v}'_M}{\partial x_j} n_i d\Gamma \\ &- \int_{\Omega} \frac{\partial \phi_k^T}{\partial x_i} \mathbf{K}_{ij}^S \frac{\partial \mathbf{v}'_M}{\partial x_j} d\Omega \\ &+ \frac{1}{2} \int_{\Omega} \phi_k^T \left(\frac{\partial \mathbf{A}_i^S}{\partial x_i} + \frac{\partial^2 \mathbf{K}_{ij}^S}{\partial x_i \partial x_j} \right) \mathbf{v}'_M d\Omega, \end{aligned} \quad (107)$$

for $k = 1, \dots, M$. A weak implementation of the boundary condition (106) amounts to substituting $\mathbf{g}(\mathbf{x}, t)$ into \mathbf{v}'_M in the boundary integrals in (107).

Doing so yields:

$$\begin{aligned} \int_{\Omega} \phi_k^T \dot{\mathbf{v}}'_M d\Omega &= -\frac{1}{2} \int_{\partial\Omega} \phi_k^T \left(\mathbf{A}_i^S + \frac{\partial \mathbf{K}_{ji}^S}{\partial x_j} \right) n_i \mathbf{g} d\Gamma \\ &+ \int_{\partial\Omega} \phi_k^T \mathbf{K}_{ij}^S \frac{\partial \mathbf{g}}{\partial x_j} n_i d\Gamma \\ &- \int_{\Omega} \frac{\partial \phi_k^T}{\partial x_i} \mathbf{K}_{ij}^S \frac{\partial \mathbf{v}'_M}{\partial x_j} d\Omega \\ &+ \frac{1}{2} \int_{\Omega} \phi_k^T \left(\frac{\partial \mathbf{A}_i^S}{\partial x_i} + \frac{\partial^2 \mathbf{K}_{ij}^S}{\partial x_i \partial x_j} \right) \mathbf{v}'_M d\Omega, \end{aligned} \quad (108)$$

for $k = 1, \dots, M$. Next, the modal decomposition $\mathbf{v}'_M = \sum_{i=1}^M v'_{M,i} \phi_i$ is substituted into (108). For a linear system of PDEs like that considered here, (108) will give rise to a ROM dynamical system of the form

$$\dot{\mathbf{v}}_M = \mathbf{A}_M \mathbf{v}_M + \mathbf{F}_M. \quad (109)$$

The implementation of the boundary condition (108) is called a “weak implementation” because \mathbf{v}'_M will only satisfy the boundary condition (106) in a weak sense. The weak implementation of other boundary conditions (e.g., a Neumann and/or Robin boundary condition) is similar to the procedure described above for the Dirichlet boundary condition.

A.4.2. Penalty method implementation of boundary conditions

The boundary condition (106) can also be implemented using a penalty method. To do this, the following boundary term is added to the right hand side of (107):

$$\tau \int_{\partial\Omega} \phi_k^T (\mathbf{v}'_M - \mathbf{g}) d\Gamma, \quad (110)$$

for each mode ϕ_k with $k = 1, \dots, M$. In (110), the parameter $\tau \in \mathbb{R}$ is a user-specified penalty parameter that controls the strength of the enforcement of (106) on $\partial\Omega$. The idea is, as $\tau \rightarrow \infty$, $\mathbf{v}'_M \rightarrow \mathbf{g}$ on $\partial\Omega$. Equation (107) together with equation (110) yields a system of the form (109). Since, with the penalty method implementation of the boundary condition, \mathbf{v}'_M will only equal \mathbf{g} on $\partial\Omega$ in a weak sense, the penalty method implementation of

a boundary condition is also a weak implementation. The implementation of other boundary conditions, e.g., a Neumann and/or Robin boundary condition, via the penalty method is analogous to the procedure for the Dirichlet boundary condition.

A.5. Linearized compressible Navier-Stokes system matrices

If the fluid vector is given by $\mathbf{q}^T = (u_1, u_2, u_3, T, \rho)$, where T and ρ denote the fluid temperature and density respectively, the matrices that appear in (27) are given by the expressions found in [8], repeated below to keep this article self-contained.

$$\begin{aligned} \mathbf{A}_1 &\equiv \begin{pmatrix} \bar{u}_1 & 0 & 0 & R & \frac{R\bar{T}}{\bar{\rho}} \\ 0 & \bar{u}_1 & 0 & 0 & 0 \\ 0 & 0 & \bar{u}_1 & 0 & 0 \\ \bar{T}(\gamma-1) & 0 & 0 & \bar{u}_1 & 0 \\ \bar{\rho} & 0 & 0 & 0 & \bar{u}_1 \end{pmatrix}, \\ \mathbf{A}_2 &\equiv \begin{pmatrix} \bar{u}_2 & 0 & 0 & 0 & 0 \\ 0 & \bar{u}_2 & 0 & R & \frac{R\bar{T}}{\bar{\rho}} \\ 0 & 0 & \bar{u}_2 & 0 & 0 \\ 0 & \bar{T}(\gamma-1) & 0 & \bar{u}_2 & 0 \\ 0 & \bar{\rho} & 0 & 0 & \bar{u}_2 \end{pmatrix}, \\ \mathbf{A}_3 &\equiv \begin{pmatrix} \bar{u}_3 & 0 & 0 & 0 & 0 \\ 0 & \bar{u}_3 & 0 & 0 & 0 \\ 0 & 0 & \bar{u}_3 & R & \frac{R\bar{T}}{\bar{\rho}} \\ 0 & 0 & \bar{T}(\gamma-1) & \bar{u}_3 & 0 \\ 0 & 0 & \bar{\rho} & 0 & \bar{u}_3 \end{pmatrix}, \\ \mathbf{G} &= \begin{pmatrix} \frac{\partial \bar{u}_1}{\partial x} & \frac{\partial \bar{u}_1}{\partial y} & \frac{\partial \bar{u}_1}{\partial z} & \frac{R}{\bar{\rho}} \frac{\partial \bar{\rho}}{\partial x} & \frac{1}{\bar{\rho}} \left(\bar{\mathbf{u}} \cdot \nabla \bar{u}_1 + R \frac{\partial \bar{T}}{\partial x} \right) \\ \frac{\partial \bar{u}_2}{\partial x} & \frac{\partial \bar{u}_2}{\partial y} & \frac{\partial \bar{u}_2}{\partial z} & \frac{R}{\bar{\rho}} \frac{\partial \bar{\rho}}{\partial y} & \frac{1}{\bar{\rho}} \left(\bar{\mathbf{u}} \cdot \nabla \bar{u}_2 + R \frac{\partial \bar{T}}{\partial y} \right) \\ \frac{\partial \bar{u}_3}{\partial x} & \frac{\partial \bar{u}_3}{\partial y} & \frac{\partial \bar{u}_3}{\partial z} & \frac{R}{\bar{\rho}} \frac{\partial \bar{\rho}}{\partial z} & \frac{1}{\bar{\rho}} \left(\bar{\mathbf{u}} \cdot \nabla \bar{u}_3 + R \frac{\partial \bar{T}}{\partial z} \right) \\ \frac{\partial \bar{T}}{\partial x} & \frac{\partial \bar{T}}{\partial y} & \frac{\partial \bar{T}}{\partial z} & (\gamma-1) \bar{\mathbf{u}} \cdot \nabla & \frac{1}{\bar{\rho}} \left(\bar{\mathbf{u}} \cdot \nabla \bar{T} + (\gamma-1) \bar{T} \nabla \cdot \bar{\mathbf{u}} \right) \\ \frac{\partial \bar{\rho}}{\partial x} & \frac{\partial \bar{\rho}}{\partial y} & \frac{\partial \bar{\rho}}{\partial z} & 0 & \nabla \cdot \bar{\mathbf{u}} \end{pmatrix}, \end{aligned} \quad (112)$$

and

$$\mathbf{K}_{11} \equiv \frac{1}{\bar{\rho} Re} \begin{pmatrix} 2\mu + \lambda & 0 & 0 & 0 & 0 \\ 0 & \mu & 0 & 0 & 0 \\ 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & \frac{\gamma\kappa}{Pr} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (113)$$

$$\mathbf{K}_{12} \equiv \frac{1}{\bar{\rho} Re} \begin{pmatrix} 0 & \lambda & 0 & 0 & 0 \\ \mu & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (114)$$

$$\mathbf{K}_{13} \equiv \frac{1}{\bar{\rho} Re} \begin{pmatrix} 0 & 0 & \lambda & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \mu & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (115)$$

$$\mathbf{K}_{21} \equiv \frac{1}{\bar{\rho} Re} \begin{pmatrix} 0 & \mu & 0 & 0 & 0 \\ \lambda & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (116)$$

$$\mathbf{K}_{22} \equiv \frac{1}{\bar{\rho} Re} \begin{pmatrix} \mu & 0 & 0 & 0 & 0 \\ 0 & 2\mu + \lambda & 0 & 0 & 0 \\ 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & \frac{\gamma\kappa}{Pr} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (117)$$

$$\mathbf{K}_{23} \equiv \frac{1}{\bar{\rho} Re} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda & 0 & 0 \\ 0 & \mu & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (118)$$

$$\mathbf{K}_{31} \equiv \frac{1}{\bar{\rho} Re} \begin{pmatrix} 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \lambda & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (119)$$

$$\mathbf{K}_{33} \equiv \frac{1}{\bar{\rho} Re} \begin{pmatrix} \mu & 0 & 0 & 0 & 0 \\ 0 & \mu & 0 & 0 & 0 \\ 0 & 0 & 2\mu + \lambda & 0 & 0 \\ 0 & 0 & 0 & \frac{\gamma\kappa}{Pr} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \quad (120)$$

The parameters appearing in the viscous stress matrices \mathbf{K}_{ij} are: the Lamé viscosity coefficients λ and μ , the thermal diffusivity κ , the Prandtl number Pr , and the Reynolds number Re .

Acknowledgements

The research presented herein was funded by Sandia National Laboratories' Laboratory Directed Research and Development (LDRD) program. Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

References

- [1] G. Serre, P. Lafon, X. Gloerfelt, C. Bailly, Reliable reduced-order models for time-dependent linearized Euler equations, *J. Comput. Phys.* **231**(15) (2012) 5176–5194.
- [2] W.J. Layton, Stable Galerkin methods for hyperbolic systems, *SIAM J. Numer. Anal.* **20**(3) (1983) 221–233.
- [3] F. Kwasniok, Empirical low-order models of barotropic flow, *J. Atmos. Sci.* **61**(2) (2004) 235–245.
- [4] M.D. Gunzburger, On the stability of Galerkin methods for initial-boundary value problems for hyperbolic systems, *Math. Comp.* **31** (139) (1977) 661–675.
- [5] D. Amsallem, C. Farhat, Stabilization of projection-based reduced order models, *Int. J. Numer. Meth. Engng.* **91** (4) (2012) 358–377.
- [6] M.F. Barone, I. Kalashnikova, D.J. Segalman, H. Thornquist, Stable Galerkin reduced order models for linearized compressible flow, *J. Comput. Phys.* **288** (2009) 1932–1946.
- [7] I. Kalashnikova, M.F. Barone, On the stability and convergence of a Galerkin reduced order model (ROM) for compressible flow with solid wall and far-field boundary treatment, *Int. J. Numer. Meth. Engng.* **83** (2010) 1345–1375.
- [8] B. Gustafsson, A. Sundstrom, Incompletely parabolic problems in fluid dynamics, *SIAM J. Appl. Math.* **35**(2) (1978) 343–357.
- [9] S. Abarbanel, D. Gottlieb, Optimal Time Splitting for Two- and Three-Dimensional Navier-Stokes Equations with Mixed Derivatives, *J. Comput. Phys.* **41** (1981) 1–33.
- [10] S. Gugercin, A.C. Antoulas, A survey of model reduction by balanced truncation and some new results, *Int. J. Control* **77**(8) (2004) 748–766.
- [11] S. Gugercin, J.-R. Li, Smith-Type Methods for Balanced Truncation of Large Sparse Systems, *Lecture Notes in Computational Science and Engineering* **45** (2005) 49–82.
- [12] B. Gustafsson, High order difference methods for time dependent PDE, Springer-Verlag, 2008.
- [13] G. Chen. Stability of Nonlinear Systems. *Encyclopedia of RF and Microwave Engineering*, Wiley, NY, 2004 (pp. 4881–4896).
- [14] B. Gustafsson, H.-O. Kreiss, J. Olinger, Time Dependent Problems and Difference Methods, Wiley-Interscience, 1995.
- [15] K.J. Astrom, R.M. Murray, Feedback systems: an introduction for scientists and engineers, Princeton University Press, 2008.
- [16] J.L. Lumley, Stochastic tools in turbulence, Academic Press: New York, 1971.
- [17] R.H. Bishop, The Mechatronics Handbook, CRC Press LLC, 2002.
- [18] H. Kimura, Chain-Scattering Approach to H-infinity Control, Springer, 1997.
- [19] P. Holmes, J.L. Lumley, G. Berkooz, Turbulence, Coherent Structures, Dynamical Systems and Symmetry, Cambridge University Press, 1996.
- [20] H.O. Kreiss, J. Lorenz, Initial-Boundary Value Problems and the Navier-Stokes Equations, Academic Press, Inc., 1989.
- [21] T. Bui-Thanh, K. Willcox, O. Ghattas, B. van Bloemen Waanders, Goal-oriented, model constrained optimization for reduction of large-scale systems, *J. Comp. Phys.* **224** (2007) 880–896.
- [22] P. Benner, M. Castillo, E.S. Quintana-Orti, G. Quintana-Orti, Parallel model reduction of large-scale unstable systems, *Advances in Parallel Computing* **13** (2004) 251–258.
- [23] C.W. Rowley, Model reduction for fluids using balanced proper orthogonal decomposition, *Int. J. Bif. Chaos* **15** (3) (2005) 997–1013.
- [24] C.W. Rowley, T. Colonius, R.M. Murray, Model reduction for compressible flows using POD and Galerkin projection, *Physica D* **189** (2004) 115–129.
- [25] L. Sirovich, Turbulence and the dynamics of coherent structures, part III: dynamics and scaling, *Q. Appl. Math.* **45** (3) (1987) 583–590.
- [26] N. Aubry, P. Holmes, J. Lumley, E. Stone, The dynamics of coherent structures in the wall region of a turbulent boundary layer, *J. Fluid Mech.* **192** (1988) 115–173.
- [27] K. Veroy, A.T. Patera, Certified real-time solution of the parametrized steady incompressible Navier-Stokes equations: rigorous reduced-bases *a posteriori* error bounds, *J. Num. Meth. Fluids* **47** (2005) 773–788.
- [28] G. Rozza. Reduced basis approximation and error bounds for potential flows in parametrized geometries. *Commun. Comput. Phys.* **9**(1) (2011) 1–48.
- [29] B. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Transactions on Automatic Control* **26** (1) (1981).
- [30] D. Amsallem, C. Farhat, On the stability of linearized reduced-order models: descriptor vs. non-descriptor form and application to fluid-structure interaction, AIAA Paper 2001-0926, *42nd AIAA Fluid Dynamics Conference & Exhibit*, New Orleans, LA (2012).
- [31] K. Willcox, J. Peraire, Balanced model reduction via the proper orthogonal decomposition, *AIAA Journal* **40** (11) (2002) 2323–2330.
- [32] E. Tadmor, Spectral Methods for Hyperbolic Problems, Lecture Notes Delivered at Ecole des Ondes, “*Méthodes numériques d’ordre élevé pour les ondes en régime transitoire*”, 1994.
- [33] M. Rathinam, L.R. Petzold, A new look at proper orthogonal decomposition, *SIAM J. Num. Anal.* **41** (5) (2003) 1893–1925.
- [34] K. Kunisch, S. Volkwein, Galerkin Proper Orthogonal Decomposition for a General Equation in Fluid Dynamics, *SIAM J. Num. Anal.* **40** (2) (2002) 492–515.
- [35] Z. Wang, I. Akhtar, J. Borggaard, Traian Iliescu, Proper

- orthogonal decomposition closure models for turbulent flows: A numerical comparison, *Comput. Methods Appl. Mech. Engrg.* (2012) 237–240.
- [36] D. Funaro, D. Gottlieb, Convergence results for pseudospectral approximations of hyperbolic systems by a penalty-type boundary treatment, *Mathematics of Computation* **196** (57) (1991) 585–596.
- [37] A. Barbagallo, D. Sipp, P.J. Schmid, Closed-loop control of an open cavity flow using reduced-order models, *J. Fluid Mech.* **641** (2009) 1–50.
- [38] B.N. Bond, L. Daniel, Guaranteed stable projection-based model reduction for indefinite and unstable linear systems, *Proceedings of the 2008 IEEE/ACM International Conference on Computer-Aided Design* (2008).
- [39] M.J. Balajewicz, E.H. Dowell, B.R. Noack, Low-dimensional modelling of high-Reynolds-number shear flows incorporating constraints from the Navier-Stokes equation, *J. Fluid Mech.* **729** (2013) 285–308.
- [40] C.W. Rowley, I. Mezic, S. Baheri, P. Schlatter, D.S. Henningson, Reduced-order models for flow control: balanced models and Koopman modes, *Seventh IU-TAM Symposium on Laminar-Turbulent Transition*, June 2009.
- [41] Y. Chahlaoui, P. Van Dooren, Benchmark examples for model reduction of linear time invariant systems, <http://www.icm.tu-bs.de/NICONET/benchmodred.html>, Mar. 2013.
- [42] T.R. Smith, Low-dimensional models of plane Couette flow using the proper orthogonal decomposition, Ph.D. thesis, Princeton University, 2003.
- [43] The MathWorks, Inc., Control Systems Toolbox User's Guide, 1992–1998.
- [44] J. Lienemann, E.B. Rudnyi, J.G. Korvink, MST MEMS model order reduction: Requirements and benchmarks, *Linear Algebra Appl.* **415**(2-3) (2006) 469–498.
- [45] W. Weaver, Jr., S.P. Timoshenko, D.H. Young, Vibration problems in engineering, Wiley, 5th Ed., 1990.
- [46] Oberwolfach benchmark collection, <http://portal.uni-freiburg.de/imteksimulation/downloads/benchmark/>, 2005.
- [47] D. Gottlieb, S.A. Orszag, *Numerical Analysis of Spectral Methods*, SIAM, 1977.
- [48] M.A. Heroux, R.A. Bartlett, V.E. Howle, R.J. Hoekstra, J.J. Hu, T.G. Kolda, R.B. Lehoucq, K.R. Long, R.P. Pawlowski, E.T. Phipps, A.G. Salinger, H.K. Thornquist, R.S. Tuminaro, J.M. Willenbring, A. Williams, K.S. Stanley, An overview of the Trilinos project, *ACM Trans. Math. Softw.* **31** (3) (2005).
- [49] B. Kirk, J.W. Peterson, R.H. Stogner, G.F. Carey, libMesh: A C++ library for parallel adaptive mesh refinement/coarsening simulations, *Eng. Comput.* **22** (3–4) (2006) 237254.
- [50] V. Sankaran, S. Menon. LES of Scalar Mixing in Supersonic Shear Layers, *Proceedings of the Combustion Institute*, **30** (2) 2835–2842 (2004).
- [51] F. Genin, S. Menon, Studies of Shock/Turbulent Shear Layer Interaction Using Large- Eddy Simulation, *Computers and Fluids*, **39** 800–819 (2010).
- [52] F. Genin, S. Menon, Dynamics of Sonic Jet Injection into Supersonic Crossflow, *J. Turbulence* **11**(4) 1–30 (2010).
- [53] W.M. Haddad, S.G. Nersesov. Stability and control of large-scale dynamical systems: A Vector Dissipative Systems Approach, Princeton University Press, 2011.
- [54] T.H. Gronwall, Note on the derivatives with respect to a parameter of the solutions of a system of differential equations, *Ann. of Math.* **20** (2) 292–296 (1919).
- [55] I. Kalashnikova, B.G. van Bloemen Waanders, S. Arunajatesan, M.F. Barone, Stabilization of Projection-Based Reduced Order Models for Linear Time-Invariant Systems via Optimization-Based Eigenvalue Reassignment, *Comput. Meth. Appl. Mech. Engrng.* **272** 251–270 (2014).
- [56] I. Kalashnikova, S. Arunajatesan, M.F. Barone, B.G. van Bloemen Waanders, J.A. Fike. Reduced Order Modeling for Prediction and Control of Large-Scale Systems. *Sandia National Laboratories Report, SAND No. 2014-4693*. Sandia National Laboratories, Albuquerque, NM (2014).
- [57] J.A. Fike. Construction of Reduced Order Models for the Non-Linear Navier-Stokes Equations using the Proper Orthogonal Decomposition (POD)/Galerkin Method. *Sandia National Laboratories Report, SAND No. 2013-7051*. Sandia National Laboratories, Albuquerque, NM (2013).
- [58] J.-R. Li, J. White. Low-Rank solutions of Lyapunov equations. *SIAM J. Matrix Anal. Appl.* **24**(1) 260–280 (2002).
- [59] J.-R. Li. Model Reduction of Large Linear Systems via Low Rank System Gramians, *Ph.D. thesis*, MIT, 2000.
- [60] P. Benner. Factorized solution of Sylvester equations with applications in control. In B. De Moor, B. Motmans, J. Willems, P. Van Dooren, and V. Blondel, editors, *Proc. Sixteenth Intl. Symp. Math. Theory Networks and Syst.*, MTNS 2004, Leuven (B), July 5-9, 2004.
- [61] V. Simoncini, Computational methods for linear matrix equations (under review), http://www.dm.unibo.it/~simoncin/public_matriceq_rev.pdf.
- [62] I. Kalashnikova, M.F. Barone. Stable and Efficient Galerkin Reduced Order Models for Non-Linear Fluid Flow, AIAA-2011-3110, *6th AIAA Theoretical Fluid Mechanics Conference*, Honolulu, HI (June 2011).